

**PERAMALAN PRODUKSI KELAPA SAWIT MENGGUNAKAN
JARINGAN SYARAT TIRUAN DENGAN METODE
BACKPROPAGATION
(Studi kasus PT.Sandabi Indah Lestari)**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Retiana Fadma Pertiwi Sinaga

NIM: 145150207111010



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

2018

PENGESAHAN

PERAMALAN PRODUKSI KELAPA SAWIT MENGGUNAKAN JARINGAN SYARAF TIRUAN
DENGAN METODE BACKPROPAGATION
(Studi Kasus PT.Sandabi Indah Lestari)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh:

Retiana Fadma Pertiwi Sinaga

NIM: 145150207111010

Skripsi ini telah diuji dan dinyatakan lulus pada
20 April 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Budi Darma Setiawan, S.Kom, M.Cs

NIP:19841015 201404 1 002

Drs. Marji, M.T

NIP: 19670801 199203 1 001

Mengetahui

Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D

NIP: 19710518 200312 1 001

IDENTITAS PENGUJI

JADWAL PELAKSANAAN UJIAN

Hari/Tanggal : 20 APRIL 2018
 Tempat : FILKOM UB / F4.1
 Waktu : 08:45 - 10:00

PENGUJI



Candra Dewi, S.Kom, M.Sc (Penguji ke I)
 NIP. 19771114 200312 2 001



Dian Eka Ratnawati, S.Si, M.Kom (Penguji ke II)
 NIP. 19730619 200212 2 001

PERNYATAAN ORISINALITAS

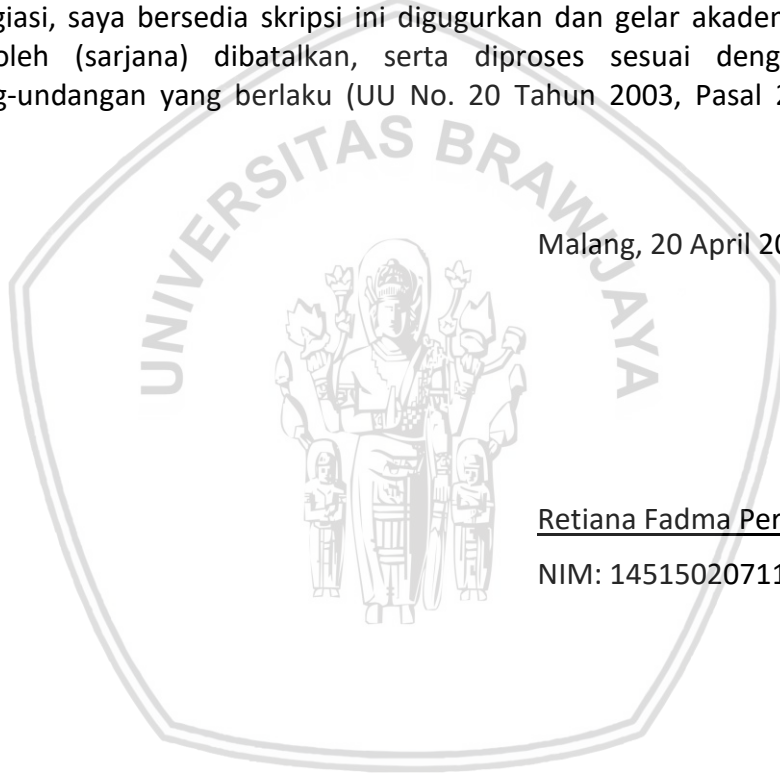
Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 20 April 2018

Retiana Fadma Pertiwi Sinaga

NIM: 145150207111010



DAFTAR RIWAYAT HIDUP

DATA PRIBADI

Nama Lengkap : Retiana Fadma Pertiwi Sinaga
Nama Panggilan : Retiana
Tempat, Tanggal Lahir: Sumatera Utara, 22 Maret 1996
Jenis Kelamin : Perempuan
Agama : Islam
Kewarganegaraan : Indonesia
Alamat Rumah : Jl. WR Supratman NO 32 kec. Muara Bangkahulu Kota Bengkulu
Alamat di Malang : Jl. Ardeli 5A lowokwaru
E-mail : retianasinaga22@gmail.com
Telepon : 087794032331



LATAR BELAKANG PENDIDIKAN

- | | |
|-------------------------------------|-----------|
| 1. TK Pertiwi Bengkulu Utara | 2001-2002 |
| 2. SDN Tomang 09 Pagi Jakarta Barat | 2002-2008 |
| 3. SMPN 18 Kota Bengkulu | 2008-2011 |
| 4. SMA PLUS N 7 Kota Bengkulu | 2011-2014 |

PENGALAMAN ORGANISASI DAN KEPANITIAAN

1. Anggota UKM Nol Derajat
2. Anggota LSO POROS 2016
3. Staff divisi Bendahara LUGOS (Linux Goes To School) 2016
4. Staff divisi PDDM BIT IF 2016

KEMAMPUAN YANG DIMILIKI

1. Kemampuan membuat program sederhana menggunakan netBeans IDE.
2. Kemampuan membuat aplikasi berbasis android.
3. Kemampuan membuat website berbahasa PHP.

KATA PENGANTAR

Puji syukur kehadiran Allah subhanahu wata'ala yang telah memberikan nikmat-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul "Peramalan Produksi Kelapa Sawit Menggunakan Jaringan Syaraf Tiruan dengan Metode Backpropagation (Studi Kasus PT. Sandabi Indah Lestari)". Dalam Penulisan skripsi ini tidak lepas dari arahan serta bibimbingan dari beberapa pihak. Maka dari itu, Penulis menyampaikan rasa terima kasih kepada:

1. Bapak Budi Darma Setiawan, S.Kom, M.Cs dan Bapak Marji, Drs., M.T selaku dosen pembimbing skripsi yang telah memberikan arahan dan membimbing penulis dalam menyelesaikan skripsi.
2. Herman Tolle, Dr. Eng., S.T, M.T selaku penasehat akademik yang memberikan saran dan masukan selama proses kuliah hingga pengerjaan skripsi ini.
3. Kedua Orang tua dan keluarga yang telah memberikan doa, nasehat serta motivasi bagi penulis hingga akhirnya penulis dapat menyelesaikan skripsi.
4. Seluruh teman seperjuangan yang telah memberi dukungan, motivasi, dan bersedia membantu penulis selama proses pengerjaan skripsi.
5. Bapak Ir. Salim Sinaga selaku Pihak PT. Sandabi Indah Lestari yang telah bersedia memberikan ilmu pengetahuan serta informasi mengenai pengolahan data yang penulis jadikan sebagai data skripsi.
6. Seluruh dosen Fakultas Ilmu Komputer Jurusan Teknik Informatika yang telah mengajarkan ilmu pengetahuan khususnya mengenai bidang teknik informatika, sehingga penulis mampu mengimplemetasikan pengetahuan yang selama ini telah di pelajari di masa perkuliahan.
7. Seluruh civitas akademik Informatika Universitas Brawijaya yang memberi pertolongan dan dukungan demi terselesaikannya skripsi ini.

Tentunya masih banyak kekurangan dari skripsi yang penulis selesaikan ini. maka dari itu, penulis sangat mengharapkan saran dan kritik yang membangun. Akhir kata semoga skripsi ini dapat digunakan dengan sebaik mungkin dan dapat memberikan manfaat bagi pembacanya.

Malang, 20 April 2018

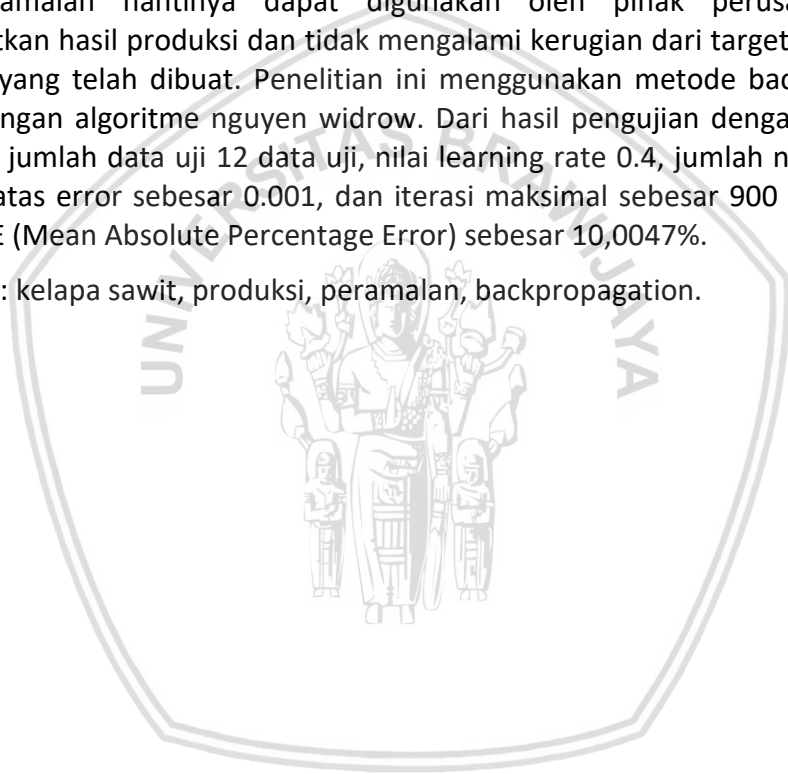
Penulis

Retianasinaga22@gmail.com

ABSTRAK

Salah satu Perkebunan Besar Swasta di Indonesia ialah PT. Sandabi Indah Lestari yang berada di Provinsi Bengkulu. PT.Sandabi Indah Lestari merancang anggaran setiap tahunnya untuk dihabiskan pada proses produksi yang dilakukan setiap seminggu sekali. Setiap proses produksi tentunya membutuhkan biaya tersendiri, jika hasil produksi tidak dapat memberikan keuntungan maka perusahaan akan mengalami kerugian. Oleh karena itu, diperlukan peramalan hasil produksi kelapa sawit untuk dijadikan acuan agar hasil produksi tetap stabil atau bahkan meningkat. Hasil peramalan nantinya dapat digunakan oleh pihak perusahaan untuk meningkatkan hasil produksi dan tidak mengalami kerugian dari target perencanaan anggaran yang telah dibuat. Penelitian ini menggunakan metode backpropagation dipadu dengan algoritme nguyen widrow. Dari hasil pengujian dengan jumlah 260 data latih, jumlah data uji 12 data uji, nilai learning rate 0.4, jumlah neuron hidden layer 5, batas error sebesar 0.001, dan iterasi maksimal sebesar 900 menghasilkan nilai MAPE (Mean Absolute Percentage Error) sebesar 10,0047%.

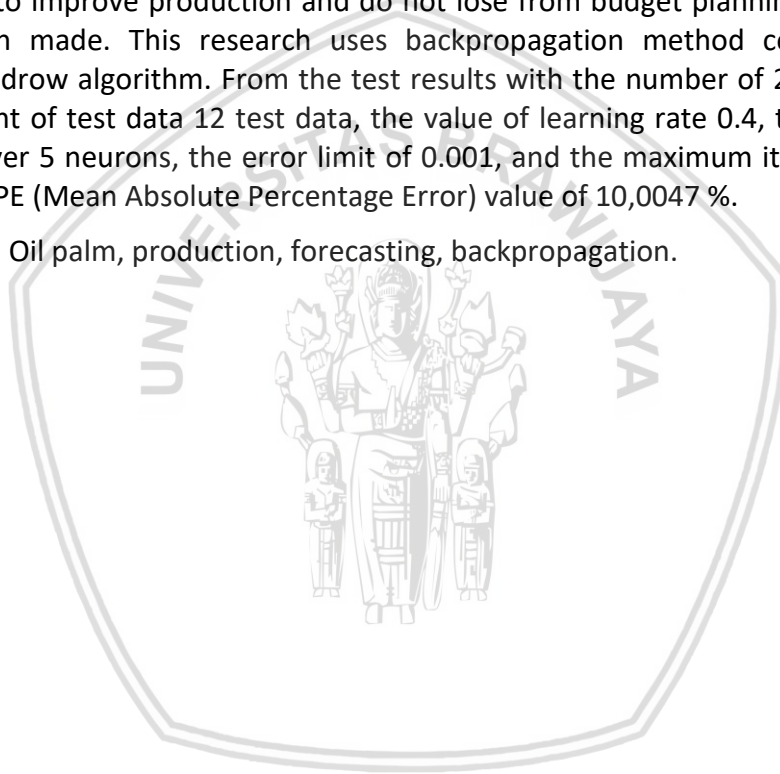
Kata kunci: kelapa sawit, produksi, peramalan, backpropagation.



ABSTRACT

One of the Big Private Plantation companies in Indonesia is PT. Sandabi Indah Lestari located in Bengkulu Province. PT.Sandabi Indah Lestari designs a budget every year to spend on production process conducted once every week. Each production process of course requires a separate cost, if the production can not change production costs, the company will incur losses. Therefore, it is necessary to forecast the output of palm oil production to be a reference for the production results remain stable or even increased. Forecasting results can later be used by the company to improve production and do not lose from budget planning targets that have been made. This research uses backpropagation method combined with nguyen widrow algorithm. From the test results with the number of 260 data train, the amount of test data 12 test data, the value of learning rate 0.4, the number of hidden layer 5 neurons, the error limit of 0.001, and the maximum iteration of 900 yields MAPE (Mean Absolute Percentage Error) value of 10,0047 %.

Keywords: Oil palm, production, forecasting, backpropagation.



DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN	xv
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan masalah.....	3
1.3 Tujuan	3
1.4 Manfaat.....	3
1.5 Batasan masalah	3
1.6 Sistematika pembahasan	4
BAB 2 LANDASAN KEPUSTAKAAN	6
2.1 Kajian Pustaka	6
2.2 Kelapa Sawit.....	7
2.3 Produksi Kelapa Sawit.....	8
2.4 Faktor-faktor Produksi Kelapa Sawit.....	9
2.4.1 Luas Lahan	9
2.4.2 Umur	9
2.4.3 Jumlah Pokok.....	9
2.4.4 Hasil Panen	10
2.5 Jaringan Syaraf Tiruan	10
2.5.1 Konsep Dasar Jaringan Syaraf Tiruan.....	11
2.5.2 Arsitektur Jaringan Syaraf Tiruan.....	11

2.5.3	Metode learning Jaringan Syaraf Tiruan	11
2.5.4	Fungsi Aktivasi Jaringan Syaraf Tiruan	12
2.6	Backpropagation	12
2.6.1	Arsitektur Dan Algoritme Backpropagation.....	13
2.6.2	Penentuan Jumlah Neuron Hidden Layer	15
2.6.3	Algoritme Nguyen-Widrow	16
2.6.4	Normalisasi dan Denormalisasi Data	16
2.6.5	Nilai Evaluasi	17
BAB 3	METODOLOGI PENELITIAN	18
3.1	Studi Literatur	18
3.1.1	Deskripsi Umum Sistem	19
3.2	Metode Pengumpulan Data	19
3.3	Perancangan Algoritme	19
3.4	Implementasi Sistem	20
3.5	Pengujian dan Analisis Hasil	20
3.6	Kesimpulan	20
BAB 4	PERANCANGAN ALGORITME	21
4.1	Formulasi Masalah	21
4.2	Flowchart	23
4.2.1	Pencarian Struktur Jaringan Terbaik	23
4.2.2	Pelatihan Data	23
4.2.3	Inisialisasi Bobot Awal	25
4.2.4	Fase Propagasi Maju	28
4.2.5	Fase Propagasi Mundur	31
4.2.6	Ubah bobot wjk , vij, wok , dan voj	35
4.3	Perhitungan Manualisasi	37
4.3.1	Normalisasi Data	37
4.3.2	Inisialisasi Bobot Awal	39
4.3.3	Fase Propagasi Maju	41

4.3.4	Fase Propagasi Mundur	43
4.3.5	Ubah bobot wjk , vij, wok , dan voj.....	44
4.3.6	Pengujian data	47
4.4	Perancangan User Interface (Antarmuka)	49
4.4.1	Perancangan Halaman Data Aktual	49
4.4.2	Perancangan Halaman Normalisasi Data.....	50
4.4.3	Perancangan Halaman Pelatihan	52
4.4.4	Perancangan Halaman Pengujian	54
4.4.5	Perancangan Halaman Peramalan Produksi	55
4.5	Perancangan Pengujian.....	57
4.5.1	Perancangan Pengujian Jumlah Iterasi Maksimal.....	57
4.5.2	Perancangan Pengujian Jumlah Data Latih	58
4.5.3	Perancangan Pengujian Nilai Learning Rate	58
4.5.4	Perancangan Pengujian Jumlah Neuron Hidden Layer	59
4.5.5	Perancangan Pengujian K-Fold Cross Validation	59
BAB 5	IMPLEMENTASI.....	61
5.1	Implementasi Sistem.....	61
5.1.1	Implementasi Normalisasi data	61
5.1.2	Implementasi Inisialisasi Bobot Awal.....	61
5.1.3	Implementasi Fase Propagasi Maju	63
5.1.4	Implementasi Fase Propagasi Mundur	64
5.1.5	Ubah Bobot	66
5.1.6	implementasi Pengujian	67
5.1.7	Implementasi Peramalan	68
5.2	Implementasi User Interface.....	69
5.2.1	Implementasi User Interface Data Aktual	69
5.2.2	Implementasi User Interface Normalisasi.....	70
5.2.3	Implementasi User Interface Pelatihan	70
5.2.4	Implemetasi User Interface Pengujian.....	71
5.2.5	Implementasi User Interface Peramalan	72

BAB 6 PENGUJIAN DAN ANALISIS HASIL	73
6.1 Pengujian	73
6.1.1 Hasil Pengujian Iterasi Maksimal	73
6.1.2 Hasil Pengujian Jumlah Data Latih	74
6.1.3 Hasil Pengujian Nilai Learning Rate	76
6.1.4 Hasil pengujian Jumlah Neuron Hidden Layer	77
6.1.5 Hasil Pengujian K-Fold Cross Validation.....	78
6.2 Analisis Hasil Pengujian	81
BAB 7 PENUTUP	82
7.1 Kesimpulan	82
7.2 Saran	83
DAFTAR PUSTAKA	84
LAMPIRAN A DATA PRODUKSI KELAPA SAWIT.....	86
LAMPIRAN B HASIL WAWANCARA	96

DAFTAR TABEL

Tabel 2.1 penelitian-penelitian sebelumnya.....	6
Tabel 2.2 Taksoomi tanama kelapa sawit	8
Tabel 3.1 Alur kerja Sistem Peramalan Produksi Kelapa Sawit.....	19
Tabel 4.1 Data Kelapa Sawit PT. Sandabi Indah Lestari	22
Tabel 4.2 Data produksi 2015 sampai 2017 PT. Sandabi Indah Lestari	37
Tabel 4.3 Data terendah dan tertinggi tiap parameter	38
Tabel 4.4 Data Produksi setelah di normalisasi	39
Tabel 4.5 Bobot vij random antara -0.5 sampai 0.5.....	40
Tabel 4.6 Bobot vij baru yang digunakan sebagai inisialisasi awal	41
Tabel 4.7 Bias awal yang ditentukan acak antara -0.92 sampai 0.92	41
Tabel 4.8 Bobot wok random antara -0.5 sampai 0.5.....	41
Tabel 4.9 Bobot wjk random antara -0.5 sampai 0.5.....	41
Tabel 4.10 Δvij dari data ke 1	44
Tabel 4.11 bobot vij baru untuk data selanjutnya	44
Tabel 4.12 Bobot vij pada data latih ke-30	45
Tabel 4.13 Bobot voj pada data latih ke-30	46
Tabel 4.14 Bobot Wjk pada data latih ke-30.....	46
Tabel 4.15 Bobot Wok pada data latih ke-30.....	46
Tabel 4.16 Perbandingan nilai target (T) dan output (Y)	46
Tabel 4.17 data uji yang sudah di normalisasi	47
Tabel 4.18 Bobot vij.....	47
Tabel 4.19 Bobot voj	47
Tabel 4.20 Bobot wjk.....	48
Tabel 4.21 Bobot wok	48
Tabel 4.22 Hasil output (Y) di badingnkan dengan target (T)	48
Tabel 4.23 Hasil Denormalisasi t dan y	48
Tabel 4.24 Halaman Data Aktual.....	49
Tabel 4.25 Halaman Normalisasi Data	51

Tabel 4.26 Halaman Pelatihan	53
Tabel 4.27 Halaman Pengujian.....	54
Tabel 4.28 Halaman Peramalan Produksi	56
Tabel 4.29 Perancangan Pengujian Jumlah Iterasi Maksimal	57
Tabel 4.30 Perancangan Pengujian Jumlah Data Latih	58
Tabel 4.31 Perancangan Pengujian Nilai Learning Rate.....	58
Tabel 4.32 Perancangan Pengujian Jumlah Neuron Hidden Layer	59
Tabel 4.33 Perancangan Pengujian K-Fold Cross Validation dengan nilai $k=5$	60
Tabel 4.34 Perancangan Pengujian K-Fold Cross Validation dengan nilai $k=7$	60
Tabel 6.1 Hasil Pengujian Jumlah Iterasi Maksimal	73
Tabel 6.2 Hasil Pengujian Jumlah Data Latih	75
Tabel 6.3 Hasil Pengujian Nilai Learning Rate	76
Tabel 6.4 Hasil Pengujian Jumlah Neuron Hidden Layer	77
Tabel 6.5 Hasil Pengujian K-Fold Cross Validation dengan nilai $k=5$	79
Tabel 6.6 Hasil Pengujian K-Fold Cross Validation dengan nilai $k=7$	79
Tabel 6.7 Analisis Hasil Pengujian	81

DAFTAR GAMBAR

Gambar 1.1 Grafik Produksi Kelapa Sawit periode 2011-2015.....	1
Gambar 2.1 Arsitektur Backpropagation	13
Gambar 3.1 Alur Metodologi Penelitian	18
Gambar 4.1 Arsitektur Jaringan Syaraf Tiruan	21
Gambar 4.2 pencarian jaringan terbaik	23
Gambar 4.3 flowchart Pelatihan Data.....	24
Gambar 4.4 Flowchart Inisialisasi Bobot Awal.....	26
Gambar 4.5 Flowchart Algoritme Nguyen Widrow	28
Gambar 4.6 flowchart Fase Propagasi Maju	28
Gambar 4.7 Sinyal Masuk Hidden Layer	30
Gambar 4.8 Sinyal Masuk Output Layer	31
Gambar 4.9 Fase Propagasi Mundur.....	32
Gambar 4.10 Koreksi Error Output Layer	33
Gambar 4.11 Koreksi Error Hidden Layer	35
Gambar 4.12 flowchart ubah bobot wjk , vij, wok , dan vok	37
Gambar 4.13 Perancangan Halaman Data Aktual	49
Gambar 4.14 Perancangan Halaman Normalisasi Data	51
Gambar 4.15 Perancangan Halaman Pelatihan	52
Gambar 4.16 Perancangan Halaman Pengujian	54
Gambar 4.17 Perancangan Halaman Peramalan	55
Gambar 4.18 Perancangan K-Fold.....	59
Gambar 5.1 User Interface Data Aktual.....	70
Gambar 5.2 User Interface Normalisasi.....	70
Gambar 5.3 User Interface Pelatihan.....	71
Gambar 5.4 User Interface Pengujian.....	72
Gambar 5.5 User Interface peramalan	72
Gambar 6.1 Pengujian Iterasi Maksimal	74
Gambar 6.2 Pengujian Jumlah Data Latih	75

Gambar 6.3 Pengujian Nilai Learning Rate	77
Gambar 6.4 Pengujian Jumlah Neuron Hidden Layer	78
Gambar 6.5 Pengujian K-fold Cross Validation dengan nilai $K=5$	80
Gambar 6.6 Pengujian K-fold Cross Validation dengan nilai $K=10$	80
Gambar 6.7 Hasil Peramalan dengan Backpropagation	81



DAFTAR LAMPIRAN

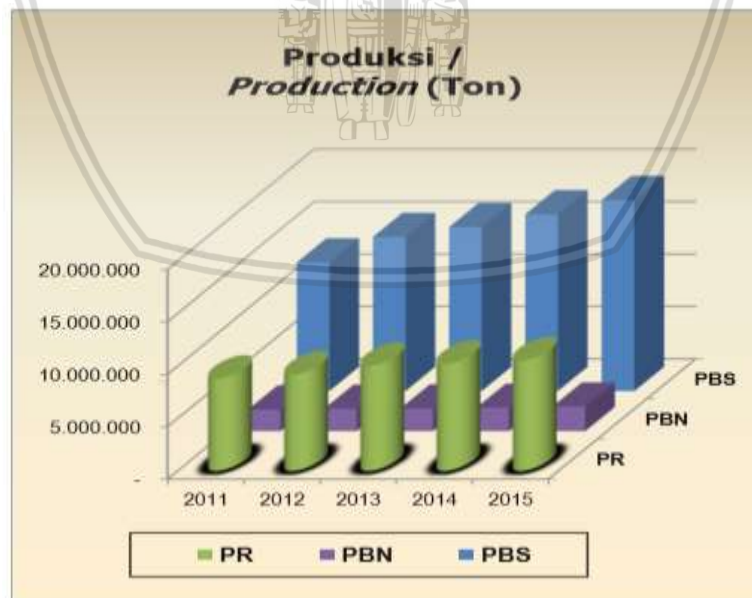
LAMPIRAN A DATA PRODUKSI KELAPA SAWIT	86
LAMPIRAN B HASIL WAWANCARA.....	96



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Perkebunan terbesar di Indonesia adalah perkebunan kelapa sawit. Awal mula pertumbuhan kelapa sawit yang cukup pesat yaitu pada tahun 1970, kemudian pada tahun 1980 mulai banyak berdirinya perkebunan kelapa sawit swasta di Indonesia. Di Indonesia terdapat tiga jenis perkebunan berdasarkan kepemilikannya, yang pertama Perkebunan Besar Negara (PBN), kemudian Perkebunan Besar Swasta (PBS) dan yang terakhir Perkebunan Rakyat (PR). Perkebunan besar Negara merupakan perkebunan yang pengolahannya dilakukan oleh pemerintah guna menjadi masukan keuangan Negara. Perkebunan besar swasta merupakan perkebunan yang dibangun oleh seorang pengusaha yang mampu dan memberikan lapangan pekerjaan untuk orang banyak. Perkebunan Rakyat memiliki pola pelaksanaan yang pengelolaannya dibantu dan di bimbing oleh Perusahaan Swasta dengan sistem *plasma*. Sistem *plasma* merupakan Pola yang bersifat bagi hasil dan menguntungkan kedua belah pihak yang juga merupakan program dari pemerintah sebagai upaya meningkatkan dan mensejahterakan pemerataan ekonomi khususnya untuk masyarakat pertanian (Badrin,2010). Hasil produksi kelapa sawit di Indonesia termasuk tinggi dibandingkan Negara penghasil kelapa sawit lainnya(Kemenperin, 2010). Gambar grafik 1.1 menunjukkan produksi kelapa sawit di Indonesia pada tahun 2012 hingga 2015 berdasarkan buku statistika perkebunan Indonesia.



Gambar 1.1 Grafik Produksi Kelapa Sawit periode 2011-2015

Sumber : (Buku Statistik Perkebunan Indonesia, 2016)

Dapat di lihat pada gambar 1.1, Produksi kelapa sawit yang paling mengalami peningkatan adalah Perkebunan Besar Swasta (PBS). Dapat disimpulkan PBS merupakan perkebunan yang berpengaruh besar dalam perindustrian kelapa sawit di Indonesia. Bahkan Pemerintah menetapkan kebijakan pengembangan bahan bakar nabati (BBN) sebagai alternatif bahan bakar minyak (BBM), dengan kebijakan tersebut memberi peluang besar bagi industri kelapa sawit untuk lebih berkembang (Kemenperin,2010). Diharapkan kedepannya tingkat produksi ini tidak menurun agar jumlah permintaan dan ketersediaan bahan baku kelapa sawit tetap seimbang. Jika kedepanya produksi kelapa sawit tidak mampu memenuhi target pengembangan BBN, maka kebijakan pemerintah menjadikan kelapa sawit sebagai alternatif bahan bakar minyak tidak tercapai kemudian harga minyak sawit akan melejit dan berpeluang besar menimbulkan dampak negatif di masyarakat karena jumlah permintaan tidak sebanding dengan hasil produksi serta segala kebutuhan pangan maupun kebutuhan pelengkap yang menggunakan bahan baku kelapa sawit akan mengalami kenaikan harga(Kemenperin,2010).

PT.Sandabi Indah Lestari merupakan salah satu contoh Perkebunan Besar Swasta(PBS) di provinsi Bengkulu. Dari hasil wawancara dengan salah satu manager perusahaan PT.Sandabi Indah lestari yaitu Ir. Salim Sinaga, Perusahaan memulai penanaman kelapa sawit pada tahun 1999 dan bertahap hingga tahun 2011. Tahap produksi di mulai pada tahun 2004 hingga sekarang. Setiap tahunnya di PT. Sandabi Indah Lestari membuat perencanaan anggaran yang akan dihabiskan untuk menangani proses produksi di perusahaan tersebut. Pada tahun 2009 PT.Sandabi Indah lestari membangun pabrik pengolahan kelapa sawitnya sendiri, sehingga Seiring berjalannya waktu permintaan bahan baku kelapa sawit naik drastis dan kegiatan produksi harus tetap stabil agar dapat memenuhi permintaan bahan baku tersebut, oleh karena itu PT. Sandabi Indah Lestari memfokuskan hasil produksi yang semaksimal mungkin agar jumlah permintaan terpenuhi dan target penentuan anggaran produksi setiap tahun tidak mengalami kerugian sehingga pendapatan keuntungan perusahaan juga tercapai. Hasil produksi sendiri dipengaruhi oleh banyak faktor, mulai dari faktor iklim, faktor lingkungan, tenaga kerja dan faktor lainnya. Oleh karena itu, diperlukan penentuan faktor yang sangat mempengaruhi hasil jumlah panen setiap bulannya agar dapat memberikan hasil peramalan yang terbaik. Penelitian ini berupa peramalan produksi kelapa sawit agar pihak perusahaan dapat menjadikan hasil peramalan sebagai acuan dalam mempertahankan kestabilan hasil produksi serta meningkatkan hasil produksi kelapa sawit.

Pada penelitian sebelumnya atas nama Eka Irawan dengan judul “Analisis Penambahan Nilai Momentum Pada Prediksi Produktivitas Kelapa Sawit Menggunakan Backpropagation” nilai MAPE(*Mean Absolute Percentage Error*) sebesar 14%, dan parameter yang digunakan dalam peramalan adalah umur tanaman, pemupukan, jumlah hari hujan, penyinaran matahari, dan defisit air (Eka Irawan,2017). Kemudian pada penelitian dengan judul “Peramalan Pasokan Tandan

Buah Segar Dan Penjualan Minyak Sawit Kasar Menggunakan Jaringan Syaraf Tiruan Propagasi Balik” menghasilkan MAPE(*Mean Absolute Percentage Error*) sebesar 25.67% dengan jenis data yaitu data penjualan kelapa sawit dari tahun 2003 hingga 2007 di PTPN XIII (Hamdani, 2012). Penulis tertarik untuk menggunakan metode *backpropagation* karena sangat cocok untuk metode forecasting (peramalan) yang dapat menghasilkan MAPE cukup baik terbukti dari penelitian-penelitian sebelumnya dan dengan menggunakan parameter yang berbeda serta jumlah data yang lebih banyak yaitu data mulai dari tahun tanam 2000 hingga 2011 dan tahun produksi mulai dari tahun 2015 hingga 2017. Parameter yang digunakan pada penelitian ini meliputi luas lahan, umur tanaman, jumlah pokok kelapa sawit, dan hasil produksi bulan sebelumnya. Maka dari itu penulis melakukan Penelitian dengan judul “Peramalan Produksi Kelapa Sawit Menggunakan Jaringan Syaraf Tiruan Dengan Metode *Backpropagation*” Studi Kasus di PT. Sandabi Indah Lestari.

1.2 Rumusan masalah

Perumusan masalah yang ingin diselesaikan pada penelitian ini antara lain:

1. Berapa nilai learning rate yang baik untuk membantu proses pelatihan data?
2. Berapa jumlah *neuron hidden layer* yang baik untuk membantu proses pelatihan data?
3. Bagaimana persentase MAPE(*Mean Absolute Percentage Error*) yang dihasilkan pada peramalan dengan menggunakan metode *backpropagation*?

1.3 Tujuan

Tujuan yang ingin dicapai dalam pembuatan skripsi ini antara lain:

1. Melakukan peramalan produksi kelapa sawit sesuai dengan faktor pendukung produksi kelapa sawit, dengan studi kasus di PT. Sandabi Indah Lestari.
2. Melakukan implementasi metode *backpropagation* kedalam aplikasi peramalan produksi kelapa sawit sehingga mampu memberikan output berupa peramalan hasil panen dalam skala ton.
3. Melakukan pengujian aplikasi Peramalan Produksi Kelapa Sawit Dengan Metode *Backpropagation*.

1.4 Manfaat

Manfaat yang diharapkan dari penelitian ini antara lain:

1. Penulis mampu menerapkan ilmu selama menuntut ilmu di FILKOM(Fakultas ilmu Komputer).
2. Memberikan informasi hasil panen kelapa sawit pada pihak perusahaan, dapat dijadikan sebagai acuan untuk meningkatkan hasil produksi dan sebagai acuan perancangan anggaran biaya produksi.

3. Dapat dijadikan sebagai referensi keilmuan oleh penelitian selanjutnya.

1.5 Batasan masalah

Batasan-batasan masalah yang menjadi pokok penelitian penulis antara lain:

1. Penelitian menggunakan jaringan syaraf tiruan dengan metode *backpropagation*.
2. Aplikasi ini difokuskan pada peramalan produksi kelapa sawit di PT. Sandabi Indah Lestari.
3. Data latih yang digunakan yaitu data produksi pada bulan januari tahun 2015 hingga bulan agustus tahun 2017 dengan parameter yang terdiri atas umur, jumlah pokok, luas lahan, hasil panen pada bulan pertama, hasil panen pada bulan kedua, hasil panen pada bulan ketiga, dan targetnya yaitu hasil panen pada bulan ke empat.

1.6 Sistematika pembahasan

BAB I. PENDAHULUAN

Bab ini menjelaskan permasalahan yang ingin diselesaikan menggunakan metode penelitian yang memiliki relasi dan mampu menyelesaikan permasalahan. Bahasan yang akan dijabarkan pada bab ini terdiri atas latar belakang masalah, identifikasi permasalahan, rumusan masalah dengan batasan-batasan masalah yang digunakan, tujuan penelitian, serta manfaat penelitian.

BAB II. LANDASAN KEPUSTAKAAN

Kajian pustaka menjelaskan teori yang mendasari penelitian, teori tersebut berasal dari penelitian terdahulu dan dari penemuan yang telah terbukti kebenarannya. Bahasan yang akan dijabarkan pada bab ini meliputi teori jaringan syaraf tiruan dengan metode *backpropagation*, teoritis tanaman kelapa sawit, sebagai acuan untuk melandasi pemecahan masalah yang sifat bahasannya bersumber dari ilmu-ilmu penelitian sebelumnya.

BAB III. METODOLOGI PENELITIAN

Pada bab ini berisi penjelasan langkah-langkah metode yang harus dilakukan dalam penelitian menggunakan jaringan syaraf tiruan dengan metode *backpropagation* dalam peramalan produksi kelapa sawit di PT. Sandabi Indah Lestari. Mulai dari studi literatur, metode pengumpulan data, perancangan algoritme, implemetasi sistem, pengujian sistem dan analisis hasil, serta kesimpulan.

BAB IV. PERANCANGAN ALGORITME

Pada bab ini berisi penjelasan tentang formulasi masalah, data pada penelitian, jumlah data yang digunakan, flowchart, perhitungan manualisasi, perancangan *user interface*, dan perancangan pengujian.

BAB V. IMPLEMENTASI

Pada bab ini berisi sourcode peramalan produksi kelapa sawit dengan metode *backpropagation* menggunakan bahasa java, penjelasan sourcode berdasarkan urutan code, dan hasil implemetasi *user interface* sesuai dengan perancangan yang telah dilakukan.

BAB VI. PENGUJIAN DAN ANALISIS HASIL

Pada bab ini metode *backpropagation* di uji kemampuannya dalam melakukan peramalan produksi kelapa sawit dengan pengujian yang terdiri atas pengujian jumlah maksimla iterasi, pengujian jumlah data latih, pengujian nilai learning rate, pengujian jumlah *neuron hidden layer* dan pengujian *k-fold cross validation*.

BAB VII. KESIMPULAN

Bab ini terdiri atas saran dan kesimpulan. Kesimpulan harus menjawab perumusan masalah serta sesuai dengan analisis hasil yang telah dipaparkan pada bab sebelumnya. Kemudian saran sebagai arahan bagi penulis untuk bahan mengoreksi kesalahan dan untuk penelitian selanjutnya.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Pada penelitian peramalan produksi kelapa sawit menggunakan metode *backpropagation* studi kasus di PT. Sandabi Indah Lestari digunakan beberapa penelitian sebelumnya sebagai dasar acuan. Penelitian dengan judul “Analisis Penambahan Nilai Momentum Pada Prediksi Produktivitas Kelapa Sawit Menggunakan *Backpropagation*” parameter yang digunakan antara lain umur tanaman, pemupukan, jumlah hari hujan dan penyinaran matahari pada 18 BSP, dan defisit air pada 24 BSP. Penelitian ini menggunakan metode *backpropagation* dengan menambahkan nilai momentum untuk pergantian nilai learning rate yang hasilnya mencapai epoch yang lebih kecil dibanding *backpropagation* tanpa penambahan nilai momentum, konvergensi tercapai pada epoch 727 dengan dan MAPE sebesar 14%. penelitian selanjutnya dengan judul “Peramalan Pasokan Tandan Buah Segar Dan Penjualan Minyak Sawit Kasar Menggunakan Jaringan Syaraf Tiruan Propagasi Balik” menghasilkan MAPE(Mean Absolute Percentage Error) sebesar 25.67% dengan jenis data yaitu data penjualan kelapa sawit dari tahun 2003 hingga 2007 di PTPN XIII. Penelitian yang terakhir dengan judul “Peramalan Dosis Pupuk Berdasarkan Karakteristik dan Lingkungan Tanaman Jeruk Siam Menggunakan Metode *Backpropagation*” menghasilkan nilai MAPE sebesar 9,178%, penelitian ini menggunakan metode *backpropagation* dengan penambahan algoritme *nguyen widrow* sebagai penentu inisialisasi bobot awal. Penjelasan Penelitian secara lebih detail dapat dilihat pada tabel 2.1

Tabel 2.1 penelitian-penelitian sebelumnya

No	Judul	Nama Peneliti	Input	Proses	Output
1	<i>Analisis Penambahan Nilai Momentum Pada Prediksi Produktivitas Kelapa Sawit Menggunakan Backpropagation</i>	Eka Irawan ,2017	umur tanaman, pemupukan, jumlah hari hujan dan penyinaran matahari pada 18 BSP, dan defisit air pada 24 BSP	Proses pelatihan menggunakan arsitektur 6-4-1 yang berarti 6 <i>neuron</i> input layer, 4 <i>neuron hidden layer</i> dan 1 <i>neuron</i> output layer. Nilai target error yang ditetapkan sebesar 0.01.	Perbandingan algoritme <i>backpropagation</i> dengan penambahan nilai momentum mencapai 727 epoch dan MSE 0,01, sedangkan algoritme <i>backpropagation</i> tanpa penambahan nilai momentum mencapai 4000 epoch dengan nilai MSE 0,001.
2	<i>Peramalan</i>	Ridhani,	lebar kanopi,	Nilai learning	komposisi

	<i>Dosis Pupuk Berdasarkan Karakteristik dan Lingkungan Tanaman Jeruk Siam Menggunakan Metode Backpropagation</i>	2017	tekstur tanah dan curah hujan.	rate yang digunakan 0.3 dengan iterasi maksimum 500, data latih 56 dan data uji 8	pupuk yakni nitrogen, fosfor, dan kalium
3	<i>Peramalan Pasokan Tandan Buah Segar Dan Penjualan Minyak Sawit Kasar Menggunakan Jaringan Syaraf Tiruan Propagasi Balik</i>	Hamdani, 2012	Data penjualan kelapa sawit di PTPN XIII	arsitektur jaringan propagasi balik yaitu 6 neuron input, 4 neuron hidden, dan 1 neuron output.	Output berupa prediksi penjualan kelapa sawit per tahun dengan hasil MAPE 25.67%.

Penelitian yang dilakukan oleh Eka Irawan (2017) memberikan saran untuk memodifikasi penentuan bobot awal dengan algoritme lain yang dapat mempercepat proses pembelajaran. Penelitian yang dilakukan oleh Ridhani (2017) memberikan saran untuk menambahkan parameter lainnya yang dapat mempengaruhi dosis pupuk serta mengkombinasikan metode *backpropagation* agar mencapai akurasi yang lebih tinggi. Penulis menentukan parameter yang akan penulis gunakan berdasarkan ketiga penelitian yang telah dijelaskan sebelumnya. penulis menggunakan parameter terdiri atas umur tanaman dengan range umur mulai dari 4 tahun hingga 17 tahun, luas lahan dalam skala hektar dengan range 5 hektar hingga 715.92 hektar, jumlah pokok kelapa sawit mulai dari 436 pokok hingga 87583 pokok, hasil panen bulan pertama, hasil panen bulan kedua, dan hasil panen bulan ketiga. Tahun tanam kelapa sawit di mulai pada periode 2000 hingga 2011 kecuali tahun 2009 karena tidak dilakukan penanaman pada tahun tersebut, serta produksi pada periode 2015 hingga 2017 setiap bulannya. Metode yang penulis gunakan adalah metode *backpropagation* dengan menambahkan algoritme penentuan bobot awal yaitu algoritme *nguyen widrow*. Algoritme nguyen widrom mampu menemukan bobot awal terbaik yang akan digunakan pada proses pelatihan agar mempercepat waktu pembelajaran *backpropagation*.

2.2 Kelapa Sawit

Tanaman kelapa sawit merupakan tanaman yang berasal dari Negara brazil. Namun pada faktanya tanaman ini sangat tumbuh dengan subur di Negara Indonesia dan beberapa Negara asia tenggara lainnya(Fauzi, 2012). Pada tingkatan

subfamily kelapa sawit termasuk dalam subfamily cocoidae, serta termasuk dalam spesies *E. Oleifera* dan *E. Odora*. Taksonomi dari tanaman kelapa sawit dapat dilihat pada tabel 2.2.

Tabel 2.2 Taksoomi tanama kelapa sawit

Divisi	Embryophyte Siphonagama
Kelas	Angiospermae
Ordo	Monocotyledoneae
Famili	aracaceae
Subfamili	Cocoidae
Genus	Elaeis
Spesies	elaeis guneensis jacq

Sumber : Fauzi(2012)

Sir Yoseph merupakan orang belanda yang pertama kali berinisiatif menanam 700 bibit di labuhan Deli, Sumatera Utara. Setelah sepuluh tahun bibit yang sudah berkembang itu di bawa ke Londen dan di ganti dengan tanaman kelapa. K. Schadt seorang berkebangsaan jerman dan M. Adrien Hallet berkebangsaan Belgia orang kedua yang mulai membuka lahan dan menanaminya dengan kelapa sawit. Kelapa sawit yang dapat dijadikan berbagai bahan kebutuhan menarik pembisnis untuk membuka perkebunan kelapa sawit karena keuntungan yang di dapat termasuk besar. Di sinilah awal mula bermunculan perusahaan kelapa sawit yang bersifat swasta (Wahyuni,2014).

2.3 Produksi Kelapa Sawit

Produksi adalah suatu proses mentransformasi usaha menjadi produk atau jasa yang berguna bagi orang banyak(irawan,2017). Produksi kelapa sawit sendiri merupakan usaha pengolahan tandan buah segar menjadi minyak murni yang nantinya dapat di olah kembali menjadi produk. Kelapa sawit belum di kategorikan sebagai tanaman yang menghasilkan minyak disebut TBM(tanaman belum menghasilkan) ,kelapa sawit mulai dapat di panen ketika umurnya telah mencapai 3 tahun (Risza, 1994). Pada era sekarang ini, permintaan kebutuhan pangan yang bahan bakunya kelapa sawit meningkat drastis. Hal ini juga membuat Indonesia harus mengimpor kelapa sawit dari Negara tetangga contohnya India. Maka dari itu hal yang harus dilakukan indoneisa ialah menyeimbangkan hasil produksi kelapa sawit dengan jumlah permintan yang ada. Banyak faktor yang mempengaruhi produksi kelapa sawit salah satunya luas lahan , iklim, jumlah pokok dan umur mempengaruhi produktivitas kelapa sawit (Risza, 1994). Pada jurnal penelitian sebelumnya, produksi kelapa sawit di di kaitkan oleh faktor iklim(irawan,2017).

2.4 Faktor-faktor Produksi Kelapa Sawit

Produksi kelapa sawit tentunya di pengaruhi oleh beberapa faktor-faktor. Banyak penelitian-penelitian terdahulu hingga sekarang yang menggali mengenai faktor yang mempengaruhi produksi kelapa sawit, mulai dari iklim, dosis pupuk, jumlah pekerja yang memanen, unsur hara tanah, umur, luas lahan, jumlah pokok dan yang lainnya. Penelitian sebelumnya yang dilakukan oleh Eka Irawan(2017) menggunakan parameter dosis pupuk, penyinaran matahari dan curah hujan sebagai faktor penentu produksi kelapa sawit, kemudian yang dilakukan oleh Edgar(2015) menggunakan parameter luas lahan, umur tanaman, jumlah hari panen, dan jumlah produksi. Pada penelitian ini, penulis menggunakan 6 parameter sebagai faktor penentu produksi kelapa sawit, diantaranya umur, luas lahan, Jumlah pokok, hasil panen bulan pertama, hasil panen bulan kedua, dan hasil panen bulan ketiga. Penjelasan mengenai faktor-faktor penentu produksi kelapa sawit yang dijadikan parameter pada penelitian ini akan dijelaskan pada sub bab selanjutnya.

2.4.1 Luas Lahan

Lahan merupakan ruang lingkup tanaman kelapa sawit di tanam. Lahan yang tepat untuk dijadikan sebagai penanaman kelapa sawit didasarkan oleh 3 faktor utama yaitu lingkungan, sifat fisik lahan, dan kesuburan tanah (Risza, 1994). Semakin luas suatu area penanaman kelapa sawit maka jumlah kelapa sawit yang di panen juga banyak. Pada penelitian ini luas lahan berdasarkan tahun tanam kelapa sawit di PT. Sandabi Indah Lestari mulai dari tahun 2000 hingga tahun 2011.

2.4.2 Umur

Umur merupakan salah satu faktor penentu produksi kelapa sawit. Ketika kelapa sawit berumur 3 tahun, proses panen mulai dilakukan. Tingkatan tertinggi hasil produksi berada pada kisaran umur 3 hingga 7 tahun, dan penurunan produktivitas kelapa sawit ketika mencapai lebih dari 15 tahun (Risza, 1994). Pada penelitian ini tanaman kelapa sawit berumur mulai dari 4 hingga 17 tahun. Berdasarkan data yang penulis teliti, ketika usia kelapa sawit melebihi umur 15 tahun, produktivitas kelapa sawit menurun, dan ketika umur 3 tahun ke atas produktivitas kelapa sawit terus meningkat.

2.4.3 Jumlah Pokok

Kelapa sawit yang baik bersumber dari bibit yang baik pula, oleh karena itu untuk mencapai produksi yang terbaik, penanaman kelapa sawit harus menggunakan bibit yang unggul (Risza, 1994). Semakin banyaknya jumlah produksi juga ditentukan oleh banyaknya jumlah pokok kelapa sawit di lahan yang tersedia untuk di tanami. Dalam 1 hektar lahan kelapa sawit produksi yang dihasilkan dapat mencapai 1 ton per bulan. Maka memaksimalkan jumlah pokok kelapa sawit yang di tanam di lahan merupakan aktivitas ppenting yang harus dilakukan dengan tujuan meningkatkan jumlah produksi.

2.4.4 Hasil Panen

Hasil panen pada bulan sebelumnya merupakan gambaran untuk hasil panen selanjutnya. Jumlah produksi dari bulan ke bulan biasanya memiliki selisih jumlah yang tidak terlalu jauh beda. Pada penelitian-penelitian sebelumnya jumlah produksi digunakan sebagai data prediksi berbagai macam studi kasus, bahkan penelitian sebelumnya dalam meramal jumlah produksi hanya menggunakan data produksi dari bulan-bulan sebelumnya saja tanpa menambahkan parameter yang lainnya.

2.5 Jaringan Syaraf Tiruan

Jaringan syaraf tiruan atau disebut juga *neural network* adalah metode yang meniru cara kerja syaraf manusia. Sel syaraf tiruan menggunakan inputan sama halnya dengan syaraf manusia yang mendapat informasi dari indra. Berikut penjelasan secara sistematis(Purnamasari,2013):

1. Sinyal yang diterima dikirimkan antara neuron atau disebut dendrit pada syaraf manusia.
2. Saat menerima sinyal dari penghubung, tiap penghubung memiliki nilai bobot. Nilai bobot mempengaruhi kuat lemahnya sinyal yang akan diterima.
3. Pada jaringan syaraf tiruan fungsi aktivasi berperan penting sebagai penentuan output baik itu output y maupun output dari z,threshold(batas ambang) memiliki peran sebagai pembanding besarnya output.

Neuron biologi memdefinisikan dua toleransi yang berlaku. Contohnya manusia dapat mengenali seseorang dari wajahnya meskipun sudah lama tidak bertemu bahkan mengenali hanya dari foto saja. Dan jika sebuah neuron rusak neuron lain dapat di latih sebagai gantinya(Siang, 2004).

Hal pokok yang jaringan syaraf tiruan lakukan ialah memorisasi dan generalisasi. Memorisasi berkaitan dengan mengenali pola-pola atau mengambil kembali pola-pola yang sudah dipelajari. Sedangkan generalisasi berkaitan dengan respon yang diberikan setelah hasil pembelajaran diharapkan respon atau keluaran sesuai dengan pola-pola sebelumnya atau ketika diberi masukan baru atau data uji yang baru, jaringan syaraf tiruan dapat mempelajari secara mandiri dari hasil output sebelumnya yang memiliki kemiripan atau mendekati(Purnamasari, 2013).

2 hal yang menjadi ciri bahwa jaringan syaraf tiruan mengikuti syaraf biologi manusia, yaitu:

1. Jaringan mendapatkan pengetahuan dari proses pembelajaranan sama halnya dengan manusia yang harus belajar terlebih dahulu untuk menguasai sebuah materi.
2. Pada neuron penghubung memiliki bobot-bobot sinaptik yang mempengaruhi kuat lemahnya dalam menyimpan pengetahuan.berikut tiga pokok mengenai jaringan syaraf tiruan:
 - a. Nama lain dari jaringan syaraf tiruan ialah pola hubungan antar neuron.
 - b. Dalam mencari bobot-bobot sinaptik menggunakan metode training atau learning dari sekumpulan input sinyal.

- c. Keluaran suatu neuron ditentukan oleh fungsi aktivasi.

2.5.1 Konsep Dasar Jaringan Syaraf Tiruan

1. Lapisan input yang menerima sinyal. Input yang diterima bisa berupa pola permasalahan yang ingin diselesaikan.
2. hidden layer(lapisan tersembunyi) layer ini tidak bisa diamati outpunya tetapi sangat berperan penting dalam pemecahan masalah yang biasanya tidak memiliki suatu target output.
3. Lapisan output merupakan hasil dari pengolahan input dan pengolahan bobot-bobot yang saling berkaitan sehingga menghasilkan solusi untuk suatu permasalahan.

2.5.2 Arsitektur Jaringan Syaraf Tiruan

Berikut arsitektur jaringan syaraf tiruan yang populer dan banyak diimplementasikan (Brian,2016):

1. Jaringan layer tunggal atau merupakan jaringan yang terdiri atas 1 lapisan input dan 1 lapisan output yang saling berhubungan. Contoh algoritme single layer network adalah Hopfield, Adaline, Perceptron. Algoritme tersebut mengolah inputan menjadi output tanpa melibatkan hidden layer.
2. Jaringan layer banyak atau di sebut juga multi layer network. Jaringan ini terdiri atas layer input, layer output serta hidden layer. Biasanya penerapannya menggunakan algoritme *backpropagation*, madeline, dan neocognitron. Proses pembelajarannya lama dikarenakan data yang kompleks, tetapi algoritme ini sangat cocok untuk memberikan solusi pada permasalahan yang kompleks.
3. Jaringan lapisan kompetitif contohnya algoritme LVQ. Jaringan ini sesuai dengan namanya yaitu kompetitif , neuron yang satu bersaing dengan neuron lainya untuk saling mendapatkan hak aktif.

2.5.3 Metode learning Jaringan Syaraf Tiruan

Berikut 3 cara pembelajaran serta pelatihan jarungan syaraf tiruan (Syaban,2016):

- a. Supervised Learning (pembelajaran terawasi). Contoh algoritme supervised learning adalah Perceptron, Hebbian, Boltzman, adaline, hopfield. Metode yang memiliki target output. Target output menjadi acuan untuk penentuan bobot bobot berikutnya, sehingga output yang dihasilkan memiliki kedekatan nilai dengan output target.
- b. Unsupervised Learning (pembelajaran tak terawasi). Contoh algoritmenya Neocognitron, Hebbian, Competitive, Kohonen, LVQ (Learning Vector Quantization). Pembelajaran tak terawasi menangani permasalahan yang tidak terdapat target output karena proses pembelajaranya untuk mendapatkan nilai bobot yang nantinya dapat dikelompokkan atau diklasifikasikan berdasarkan kedekatan nilai.

- c. Kombinasi supervised dan unsupervised adalah hybrid learning. Contoh algoritmenya adalah RBF. Biasanya metode ini digunakan dalam pengenalan gambar dan mengenali teks. Cara kerjanya sebagian menggunakan supervised learning dan sebagian lagi unsupervised learning.

2.5.4 Fungsi Aktivasi Jaringan Syaraf Tiruan

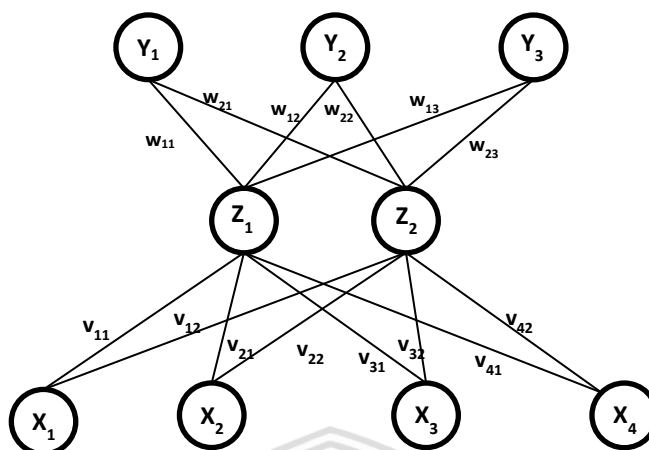
Dalam jaringan syaraf tiruan, fungsi aktivasi digunakan untuk mendefinisikan nilai output dari sebuah neuron. berikut fungsi aktivasi jaringan syaraf tiruan (Purnamasari, 2013):

- a. Fungsi threshold berperan dalam penentuan batas ambang. Threshold bersifat biner, ketika data bipolar maka threshold 0 diganti dengan -1. Disamping threshold, bias yang nilainya selalu 1 berhubungan dengan threshold untuk mengubah nilainya menjadi 0.
- b. fungsi yang nilainya bersifat diferensi adalah fungsi sigmoid.
- c. Ketika output yang dihasilkan adalah nilai yang buka dari range 0 dan 1 atau 1 dan -1 maka kita membutuhkan fungsi identitas.

2.6 Backpropagation

Backpropagation salah satu metode Jaringan Syaraf Tiruan Merupakan algoritme yang berusaha menghasilkan error minimum pada setiap output (Purnamasari, 2013). *Backpropagation* Melatih jaringan untuk tetap seimbang dalam merespon inputan yang berbeda dari data latih sebelumnya. Sehingga output yang dihasilkan tetap berhubungan dengan data latih. *Backpropagation* memiliki satu atau lebih unit pada layer output dan hidden layer. Algoritme ini menggunakan istilah umpan maju dalam mengatur bobot-bobotnya. Pelatihanya menggunakan gradien dari fungsi aktivasi, arah dari gradien ialah negatif untuk membuat fungsi aktivasi turun sehingga bobot lebih mudah diperbaiki. (Purnamasari, 2013).

2.6.1 Arsitektur Dan Algoritme *Backpropagation*



Gambar 2.1 Arsitektur *Backpropagation*

Sumber: Fausett(1994)

Secara umum arsitektur *backpropagation* dapat di lihat pada gambar 2.1. Unit input disimbolkan dengan x kemudian unit hidden layer disimbolkan dengan z dan unit output disimbolkan dengan y . penentuan bobot dari input ke hidden layer dilambangkan dengan v dan bobot dari hidden layer ke output dilambangkan dengan w . Pada *backpropagation* terdapat 3 fase perhitungan yaitu fase propagasi maju(feedforward), Fase propagasi mundur(backward), dan fase perubahan bobot. Fase tersebut di ulang secara terus menerus hingga kondisi berhenti terpenuhi, biasanya kondisi berhenti ditentukan oleh nilai maksimal iterasi atau ketika nilai y mengalami konvergen. Berikut notasi yang digunakan pada algoritme *backpropagation*:

- x : input
- t : target
- α : *learning rate*
- x_i : neuron input ke- i
- z_i : *neuron hidden* ke- i
- y_k : neuron output ke- k
- v_{ij} : bobot antara nilai neuron input dan *neuron hidden*
- v_{oj} : bobot bias neuron input ke- j
- w_{ok} : bobot bias untuk *neuron hidden* ke- k
- w_{jk} : bobot antara *neuron hidden* dan neuron output
- δ_k : faktor koreksi bobot w_{jk}
- δ_j : faktor koreksi bobot v_{ij}

Algoritme *backpropagation* sendiri di mulai dengan menginputkan data, melakukan proses propagasi maju, propagasi mundur yang merupakan perhitungan error dan pembaruan bobot serta bias hingga menemukan hasil

output yang mendekati target dan error yang bernilai minimum. Berikut langkah-langkah perhitungannya (Fausett,1994):

1. inisialisasi bias(v) dan bobot(w) dengan nilai acak antara -0.5 sampai 0.5. inisialisasi juga nilai *learning rate* (α), jumlah iterasi maksimum, serta batasan nilai error.
2. Proses inisialisasi terus dilakukan hingga kondisi berhenti terpenuhi. Kondisi berhenti ditentukan oleh nilai iterasi maksimum, nilai batasan error.

Fase Propogasi Maju

3. Tiap sinyal yang masuk pada neuron input(x) di teruskan ke semua neuron di *hidden layer*(z).
4. Ketika sinyal memasuki *hidden layer* maka hitung sinyal input dan biasnya dengan masing-masing bobot menggunakan persamaan 2.1:

$$zin_j = vo_j + \sum_{k=0}^n x_i v_{ij} \quad (2.1)$$

5. Kemudian hitung sinyal output dari *hidden layer* dengan menggunakan fungsi aktivasi sigmoid . yang nantinya sinyal ini diteruskan ke output layer, gunakan persamaan 2.2:

$$z_j = f(zin_j) = \frac{1}{1 + \exp^{-zin_j}} \quad (2.2)$$

6. Ketika sinyal memasuki output layer, maka sinyal input dan bias yang berasal dari *hidden layer* di hitung dengan menggunakan persamaan 2.3:

$$yin_k = wo_k + \sum_{j=0}^p z_j w_{jk} \quad (2.3)$$

7. hitung sinyal output dari unit output(y) dengan menggunakan persamaan 2.4:

$$y_k = f(yin_k) \quad (2.4)$$

Fase Propagasi Mundur

8. hitung faktor koreksi error output, koreksi error output merupakan error yang dihasilkan atas kesalahan unit keluaran y_k menggunakan persamaan 2.5:

$$\delta_k = (t_k - y_k)y_k(1 - y_k) \quad (2.5)$$

9. kemudian hitung juga faktor koreksi error w_{jk} dengan menggunakan persamaan 2.6:

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (2.6)$$

10. hitung nilai koreksi bias dengan menggunakan persamaan 2.7:

$$\Delta w_{ok} = \alpha \delta_k \quad (2.7)$$

11. hitung nilai koreksi delta bobot w_{jk} sebagai nilai bobot yang nantinya di kirim ke neuron di *hidden layer* dengan menggunakan persamaan 2.8:

$$\delta in_j = \sum_{k=1}^m \delta_j w_{jk} \quad (2.8)$$

12. selanjutnya nilai δin_j dikalikan dengan turunan dari fungsi aktivasi yang bertujuan untuk menghasilkan δ_j menggunakan persamaan 2.9:

$$\delta_j = \delta in_j z_j (1 - z_j) \quad (2.9)$$

13. kemudian menghitung nilai koreksi error Δv_{ij} yang berfungsi untuk mengupdate v_{ij} dengan menggunakan persamaan 2.10:

$$\Delta v_{ij} = \alpha \delta_j x_i \quad (2.10)$$

14. menghitung nilai koreksi error Δv_{oj} dengan menggunakan persamaan 2.11:

$$\Delta v_{oj} = \alpha \delta_j \quad (2.11)$$

Fase Ubah Bobot

15. hitung perubahan bobot w_{jk} dengan menggunakan persamaan 2.12:

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \quad (2.12)$$

16. hitung perubahan bobot v_{ij} dengan menggunakan persamaan 2.13:

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad (2.13)$$

17. hitung perubahan bobot bias v_{oj} dengan menggunakan persamaan 2.14:

$$v_{oj}(\text{baru}) = v_{oj}(\text{lama}) + \Delta v_{oj} \quad (2.14)$$

18. hitung perubahan bobot w_{ok} dengan menggunakan persamaan 2.12:

$$w_{ok}(\text{baru}) = w_{ok}(\text{lama}) + \Delta w_{ok} \quad (2.15)$$

19. Setelah pelatihan dilakukan, periksa kondisi berhenti yakni apabila nilai error sudah lebih kecil dari nilai batas yang telah ditentukan, atau kondisi berhenti juga dapat ditentukan dari maksimum iterasi. perhitungan error pada pelatihan menggunakan persamaan 2.16:

$$MSE = \sum_i^n \frac{(t-y)^2}{n} \quad (2.16)$$

keterangan :

y = nilai ramalan

t = nilai data aktual

n = jumlah data

2.6.2 Penentuan Jumlah Neuron Hidden Layer

Jumlah *neuron hidden layer* mempengaruhi keakuratan algoritme *backpropagation* pada tahap pelatihan, maka diperlukan penentuan jumlah *hidden layer* yang sesuai agar pelatihan yang dilakukan menghasilkan output(y) yang mendekati nilai target. Berikut aturan penentuan jumlah *hidden layer* (Purnamasari,2013):

- Jumlah *neuron hidden layer* nilainya berada di antara nilai neuron input layer dan neuron output layer.
- Jumlah *neuron hidden layer* adalah 2/3 dari jumlah neuron input layer dan ditambah dengan jumlah neuron output layer.

- c. Banyaknya neuron di *hidden* layer kurang dari dua kali jumlah neuron di input layer.

2.6.3 Algoritme Nguyen-Widrow

Algoritme *nguyen widrow* digunakan untuk menentukan inisialisasi bobot awal terbaik yang akan digunakan pada proses pelatihan *backpropagation*. Inisialisasi ini diharapkan dapat mengurangi waktu pelatihan, berikut tahapan dari algoritme *nguyen-widrow* (Syaban,2016):

1. Inisialisasi nilai w_{ok} , w_{jk} , v_{oj} , v_{ij} . Nilai acak antara -0.5 sampai 0.5.

2. Kemudian cari nilai $||v_{ij}||$ menggunakan persamaan 2.17:

$$||v_{ij}|| = \sqrt{v_{1j}^2 + v_{2j}^2 \dots v_{nj}^2} \quad (2.17)$$

3. Kemudian hitung faktor skala menggunakan persamaan 2.18:

$$\beta = 0.7^n \sqrt{p} \quad (2.18)$$

Keterangan :

n = jumlah unit input

p = jumlah unit *hidden*

4. Kemudian nilai v_{ij} yang akan digunakan pada inisialisasi bobot awal didapatkan dengan menggunakan persamaan 2.19:

$$v_{ij} = \frac{\beta v_{ij}(\text{lama})}{||v_{ij}||} \quad (2.19)$$

5. Nilai bias yang akan digunakan pada inisialisasi adalah v_{0j} = bilangan acak antara $-\beta$ sampai β .

6. Tentukan Nilai batasan w_{ok} dan w_{jk} antara -0.5 samapai 0.5.

2.6.4 Normalisasi dan Denormalisasi Data

Pada penelitian ini penulis menggunakan fungsi aktivasi sigmoid biner, yang nilai rangenya antara 0 sampai 1. Data asli perlu dilakukan normalisasi agar range data berada di berada antara 0 sampai 1 sesuai dengan nilai aktivasinya. Perhitungan normalisasi menggunakan persamaan 2.20 (Siang, 2009):

$$y = \frac{x - \min}{\max - \min} \cdot 0.8 + 0.1 \quad (2.20)$$

Keterangan :

y = nilai hasil normalisasi

x = nilai sebelum proses normalisasi

\min = nilai terendah di dalam data

\max = nilai tertinggi di dalam data

Setelah data di normalisasi untuk mengembalikan nilai data yang sebenarnya dilakukan denormalisasi data. perhitungan denormalisasi menggunakan persamaan 2.21:

$$y' = \frac{x' - 0.1}{0.8} (\max - \min) + \min \quad (2.21)$$

y' = nilai denormalisasi

x' = nilai sebelum proses denormalisasi

min = nilai terendah di dalam data
max = nilai tertinggi di dalam data

2.6.5 Nilai Evaluasi

Nilai Evaluasi digunakan sebagai pengukur tingkat keakuratan yang dihasilkan pada proses peramalan, nilai ini dalam bentuk persentase. Pada penelitian nilai evaluasi yang digunakan adalah *Mean Absolute Percentage Error* (MAPE). Evaluasi menggunakan MAPE digunakan pada tahap pengujian berdasarkan hasil peramalan dengan nilai aktual. Rumus MAPE dapat dilihat pada persamaan 2.22 (Puspita, 2007):

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y-t}{y} \right| \times 100 \quad (2.22)$$

y = hasil ramalan

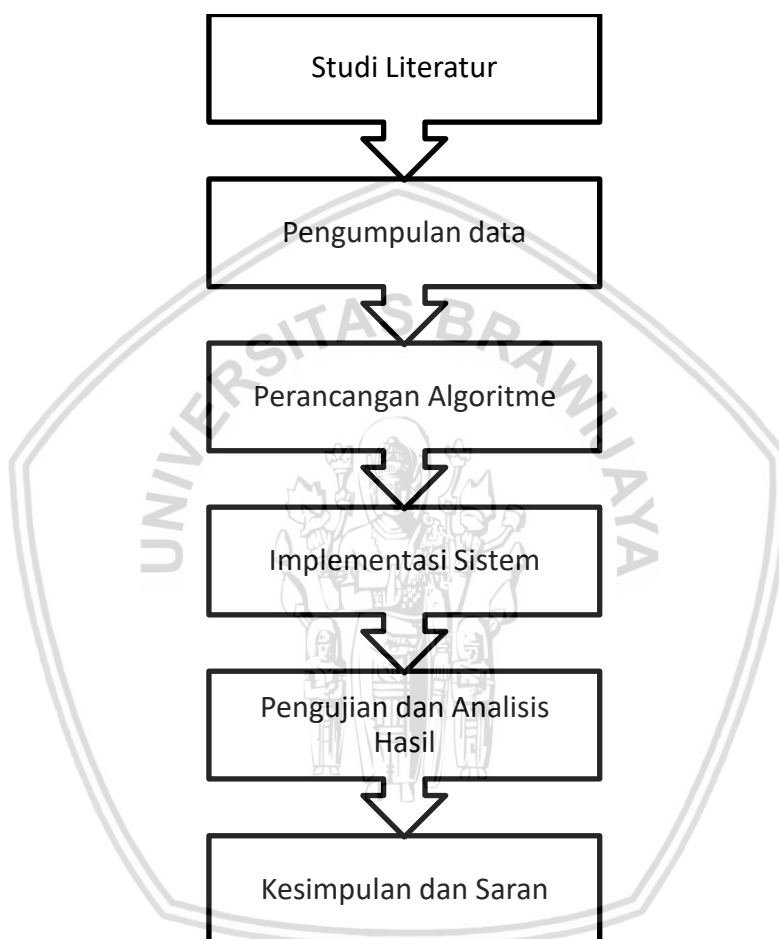
t = nilai aktual

n = jumlah data



BAB 3 METODOLOGI PENELITIAN

Dalam melakukan penelitian, tahapan-tahapan metode dalam meneliti suatu permasalahan harus dijabarkan. Penelitian yang penulis lakukan yaitu peramalan produksi kelapa sawit di PT. Sandabi Indah Lestari merupakan jenis penelitian yang melibatkan pengolahan data, Berikut diagram alir metode penelitian yang penulis lakukan dapat di lihat pada gambar 3.1.



Gambar 3.1 Alur Metodologi Penelitian

3.1 Studi Literatur

Studi literatur dari berbagai karya ilmiah adalah metode yang digunakan dalam penelitian sebelumnya yang berkaitan dengan peramalan produksi tanaman kelapa sawit dan tanaman lainnya, berdasarkan penelitian sebelumnya akurasi tertinggi didapatkan dengan menggunakan metode *backpropagation* sehingga penulis memutuskan untuk menggunakan metode *backpropagation* dan implementasi programnya menggunakan bahasa java.

3.1.1 Deskripsi Umum Sistem

Perangkat lunak yang akan di buat pada penelitian ini ialah perangkat lunak untuk peramalan hasil produksi kelapa sawit menggunakan metode *backpropagation* dengan Studi kasus di PT. Sandabi Indah Lestari Kota Bengkulu. Sistem ini dapat digunakan oleh *user* untuk meramal hasil produksi pada bulan selanjutnya dengan menginputkan parameter luas lahan, umur, jumlah pokok dan produksi bulan pertama, kedua dan ketiga. Output yang dihasilkan oleh perangkat lunak ini berupa peramalan hasil produksi bulan selanjutnya.

3.2 Metode Pengumpulan Data

Berikut dua metode yang digunakan dalam proses penelitian dan pengumpulan data pendukung:

1. Wawancara

Wawancara merupakan metode tanya jawab antara penanya dan narasumber. Wawancara yang penulis lakukan bertujuan mendapatkan informasi produksi dan data yang terkomputasi di PT. Sandabi Indah Lestari. Penulis melakukan wawancara dengan seorang narasumber dari perusahaan PT. Sandabi Indah Lestari yaitu Ir. Salim Sinaga Selaku manager perkebunan. Penulis menggali informasi mengenai sejarah dan prosedur produksi di PT. sandabi Indah Lestari serta data produksi yang telah terkomputasi. Pertanyaan dan jawaban akan dilampirkan pada Lampiran B.

2. Studi Pustaka

Merupakan metode Pencarian informasi berupa data-data pendukung bahan penelitian yang di ambil dari tulisan-tulisan yang berada pada buku, jurnal ilmiah, makalah serta artikel yang bersifat penelitian ilmiah. Hasilnya nanti dikumpulkan menjadi data yang membahas penelitian studi pustaka berupa teori *backpropagation*, kelapa sawit dan teori pendukung lainnya.

3.3 Perancangan Algoritme

Perancangan Algoritme pada penelitian ialah perancangan metode *backpropagation* dalam menyelesaikan permasalahan peramalan produksi kelapa sawit di PT. Sandabi Indah Lestari. Sistem juga menggunakan database untuk menampung data latih yang berisi sekumpulan parameter yang digunakan untuk meramal hasil produksi. Perancangan sistem berdasarkan input, proses, dan output dijelaskan pada tabel 3.1.

Tabel 3.1 Alur kerja Sistem Peramalan Produksi Kelapa Sawit

Input	Proses	Output
Umur tanaman, jumlah pokok, luas lahan, hasil panen bulan pertama, hasil panen bulan kedua, dan hasil panen bulan ketiga. Kemudian menginputkan jumlah data uji	Melakukan proses pelatihan data yang di mulai dari inisialisasi bobot awal, propagasi maju, propagasi mundur, ubah bobot. Kemudian dilakukan pengujian.	Nilai MSE, Iterasi, MAPE, Akurasi, dan hasil peramalan dalam skala ton

3.4 Implementasi Sistem

Implementasi dilakukan setelah proses perancangan. Tahapan ini mengimplementasikan rancangan Algoritme ke dalam coding yang menghasilkan perangkat lunak untuk menyelesaikan masalah pada penelitian ini. Dalam mengimplementasikan sendiri kebutuhan pendukung baik itu perangkat lunak dan perangkat keras harus sudah terpenuhi.

3.5 Pengujian dan Analisis Hasil

Pengujian sistem dilakukan ketika perangkat lunak sudah jadi dan perlu di uji coba terlebih dahulu. Pengujian ini meliputi fungsi-fungsi, hasil output dan algoritme yang digunakan. Pada penelitian ini pengujian perangkat lunak dilakukan untuk menguji algoritme *backpropagation* yang telah di implementasikan. Pengujian ini bertujuan untuk memastikan bahwa metode *backpropagation* sudah benar atau belum dan juga mampu melakukan peramalan yang diharapkan. Jika terjadi kesalahan dalam proses pengujian maka sistem dapat dibenahi. Analisis hasil dilakukan setelah melakukan pengujian. MAPE yang dihasilkan pada saat pengujian merupakan acuan analisis dalam menentukan akurasi serta kekurangan dari aplikasi. Analisis ini nantinya dijadikan sebagai pencapaian solusi yang diharapkan serta bahan perbaikan atau pun sebagai acuan pengembangan lanjut untuk penelitian kedepannya.

3.6 Kesimpulan

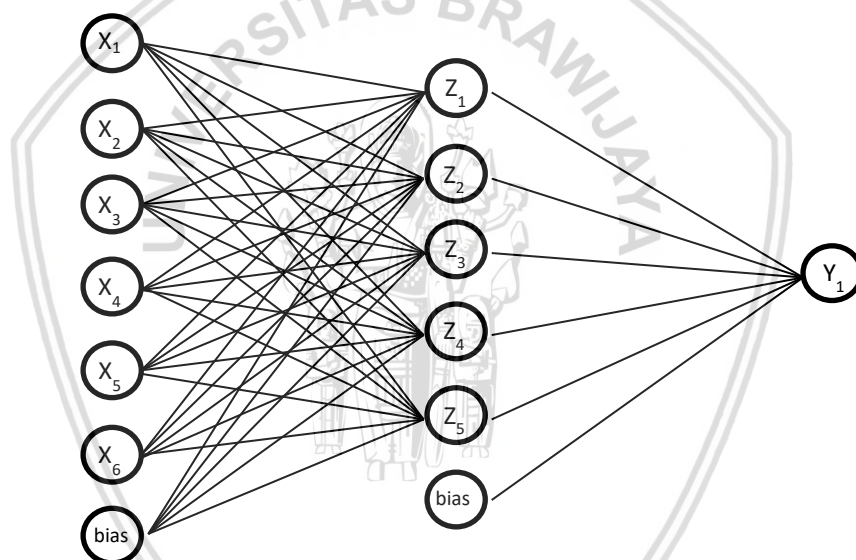
Kesimpulan akan memaparkan analisi hasil pengujian yang telah dilakukan, kekurangan dari penelitian yang dilakukan dan saran untuk penelitian selanjutnya. Hasil pengujian sistem berupa Nilai MAPE yang sudah baik atau tidak dan memenuhi perancangan yang telah dilakukan. Jika masih terdapat kesalahan maka dapat dilakukan penyempurnaan lebih lanjut untuk penelitian selanjutnya.

BAB 4 PERANCANGAN ALGORITME

Bab ini berisi formulasi masalah, flowchart, perhitungan manualisasi, perancangan *user interface* dan perancangan pengujian.

4.1 Formulasi Masalah

Pada penelitian ini permasalahan yang ingin di selesaikan adalah menentukan hasil produksi bulan berikutnya berdasarkan 6 parameter yaitu umur, luas lahan, jumlah pokok, 3 bulan hasil produksi sebelumnya. Penyelesaian permasalahan bertujuan untuk meningkatkan lagi hasil produksi dari bulan ke bulan agar dapat mencapai target keuntungan dan memenuhi permintaan. Proses peramalan produksi menggunakan algoritme *backpropagation* dan dalam menginisialisasi bobot awal terbaik digunakan algoritme *nguyen widrow* untuk digunakan pada tahap pelatihan pada *backpropagation*. Rancangan arsitektur *backpropagation* pada penelitian ini dapat dilihat pada gambar 4.1.



**Gambar 4.1 Arsitektur Jaringan Syaraf Tiruan
backpropagation pada penelitian**

Pada gambar 4.1 terdapat 6 fitur sebagai neuron input yaitu umur (x_1), jumlah pokok (x_2), luas lahan (x_3), Hasil panen bulan ke-1 (x_4), Hasil panen bulan ke-2 (x_5), dan Hasil panen bulan ke-3 (x_6). Kemudian output layer pada penelitian ini ialah peramalan hasil produksi bulan selanjutnya (y_1). Jumlah data yang digunakan pada penelitian ini ialah berjumlah 319 data, dengan jumlah 260 data digunakan sebagai data latih dan 59 data digunakan sebagai data uji. Berikut 30 data latih pada tabel 4.1 yang akan penulis jabarkan manualisasinya pada bab ini.

Tabel 4.1 Data Kelapa Sawit PT. Sandabi Indah Lestari

No	Umur (tahun)	jumlah pokok	luas lahan (hektar)	Hasil panen bulan ke-1(ton)	Hasil panen bulan ke-2(ton)	Hasil panen bulan ke-3(ton)	Target (ton)
1	16	2592	18	38	34	55	33
2	17	2592	18	34	32	40	29
3	15	33935	280	601	533	522	557
4	16	33935	280	438	488	590	487
5	14	21799	185	362	247	297	307
6	15	21799	185	240	245	286	288
7	13	87583	716	1151	1350	1241	1097
8	14	87583	716	1009	998	1174	1180
9	12	85510	700	1177	1222	1119	1110
10	13	85510	700	988	885	1138	1063
11	11	17970	149	289	244	268	252
12	12	17970	149	187	200	283	244
13	10	10583	92	136	157	134	124
14	11	10583	92	124	116	146	124
15	9	436	5	4	4	5	2
16	10	436	5	3	2	3	3
17	7	17352	157	215	193	142	140
18	8	17352	157	191	169	217	182
19	6	1339	15	13	13	12	13
20	7	1339	15	16	13	20	14
21	5	2575	13	10	13	10	9
22	6	2575	13	17	15	16	12
23	16	2592	18	34	55	33	35
24	17	2592	18	32	40	29	33
25	15	33935	280	533	522	557	469
26	16	33935	280	488	590	487	440
27	14	21799	185	247	297	307	255
28	15	21799	185	245	286	288	248
29	13	87583	716	1350	1241	1097	1149
30	14	87583	716	998	1174	1180	979

Parameter yang terdapat pada data produksi kelapa sawit PT. Sandabi Indah Lestari terdiri atas 6 parameter yaitu umur, luas lahan, jumlah pokok, hasil panen bulan pertama, hasil panen bulan kedua, dan hasil panen bulan ketiga. Berikut penjelasan dari setiap parameter di atas:

1. Umur menjelaskan usia dari tanaman kelapa sawit dalam skala tahun
2. Luas lahan menjelaskan luas dalam skala hektar yang di tanami kelapa sawit.
3. Jumlah pokok menjelaskan banyaknya tanaman kelapa sawit yang di tanam pada luas tertentu.
4. Hasil panen bulan ke 1, 2, dan 3 menjelaskan hasil produksi kelapa sawit mulai dari 3 bulan sebelumnya, contohnya hasil panen bulan januari,

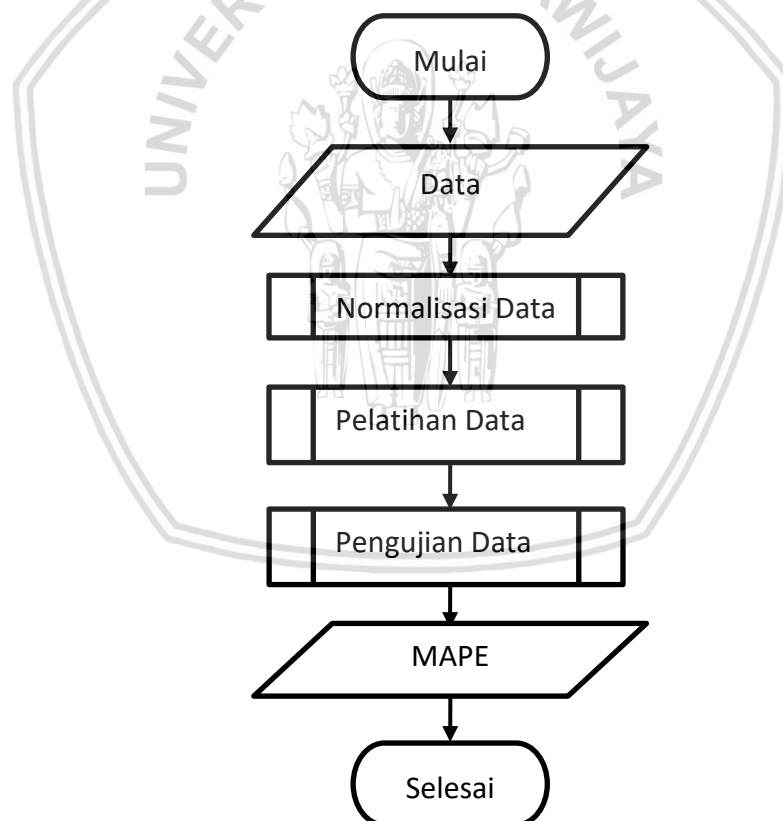
februari dan maret merupakan parameter input maka bulan ke 4 yaitu april merupakan target ramalan.

4.2 Flowchart

Flowchart adalah alur kerja pengerjaan suatu algoritme yang di gambarkan dengan bagan dan simbol-simbol. Pada penelitian ini, flowchart yang di bangun ialah flowchart untuk peramalan produksi kelapa sawit dengan menggunakan metode *backpropagation*, maka alur kerja yang di jelaskan merupakan alur kerja algoritme *backpropagation*.

4.2.1 Pencarian Struktur Jaringan Terbaik

Pencarian jaringan terbaik merupakan pencarian struktur jaringan algoritme *backpropagation* yang menghasilkan ramalan terbaik. Pencarian akan terus dilakukan untuk mendapatkan struktur jaringan terbaik dengan cara menentukan batasan nilai MSE yang sekecil mungkin dan nilai MAPE yang rendah dalam peramalan produksi kelapa sawit. Flowchart pencarian jaringan terbaik dapat dilihat pada gambar 4.2.

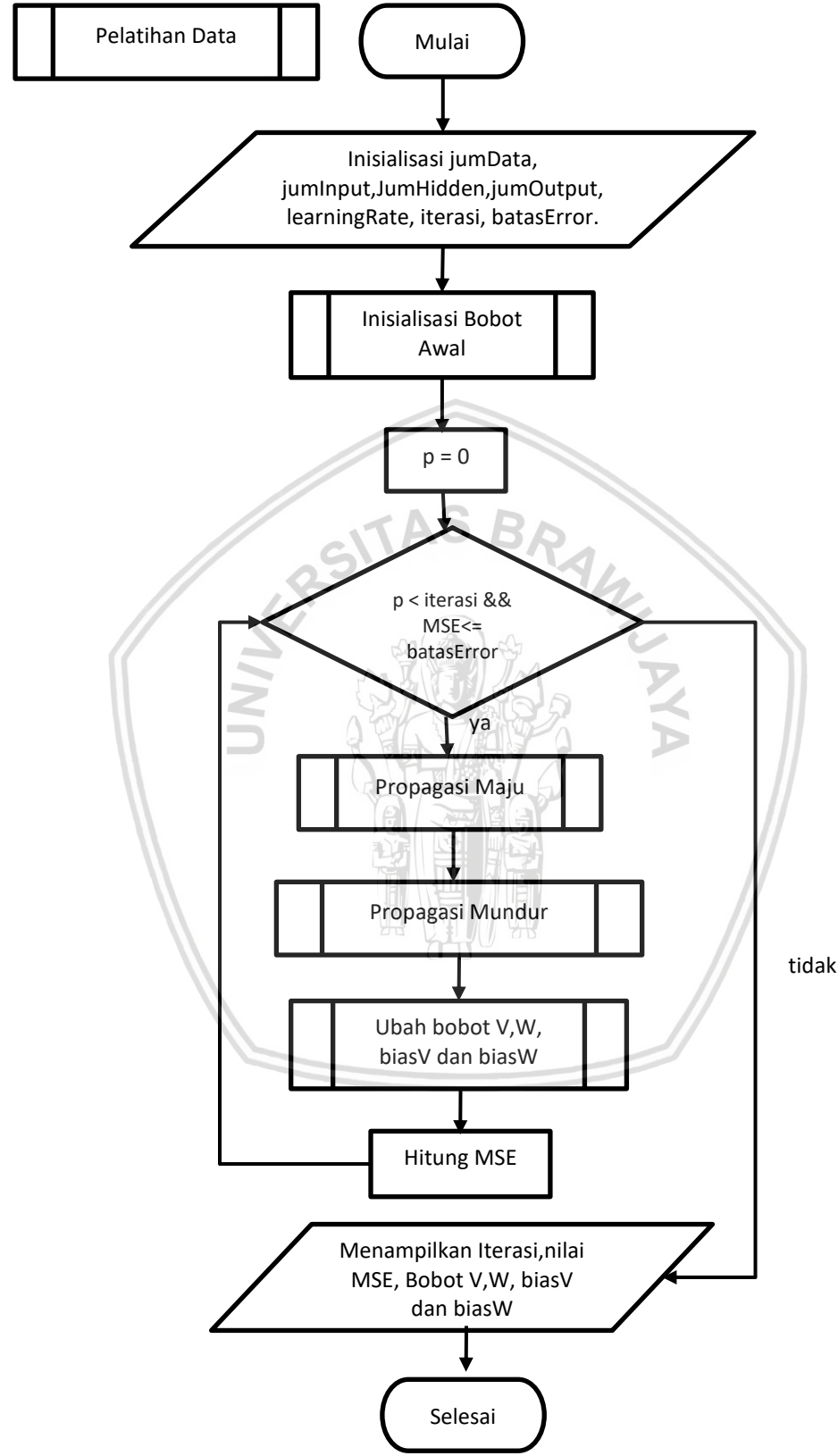


Gambar 4.2 pencarian jaringan terbaik

4.2.2 Pelatihan Data

Pelatihan data dilakukan untuk mendapatkan struktur jaringan terbaik dalam peramalan kelapa sawit. Pelatihan terdiri atas 3 fase yaitu fase propagasi maju, fase propagasi mundur dan mengubah bobot serta bias. Flowchart 4.3

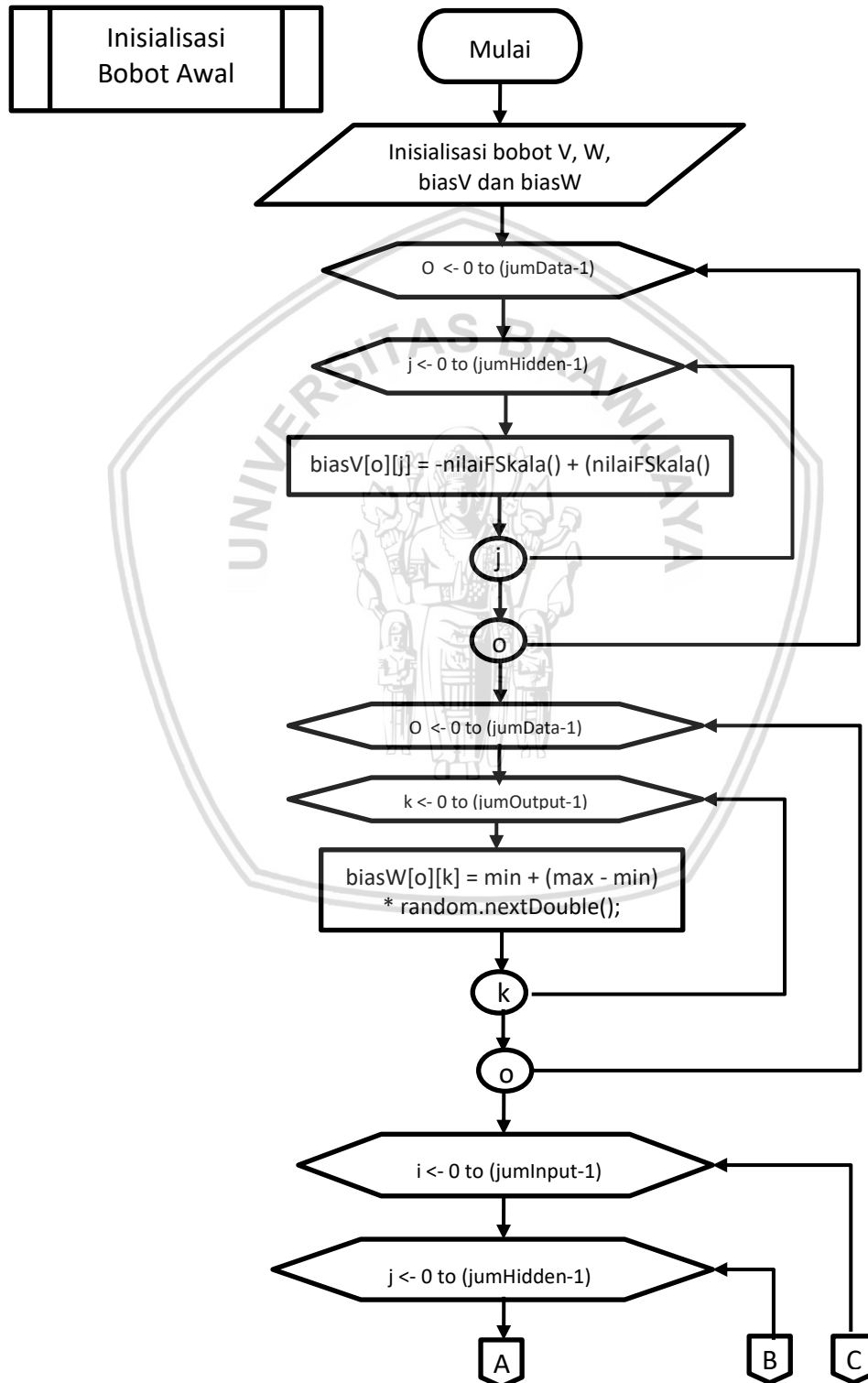
menjelaskan alur kerja dari pelatihan data *backpropagation*.

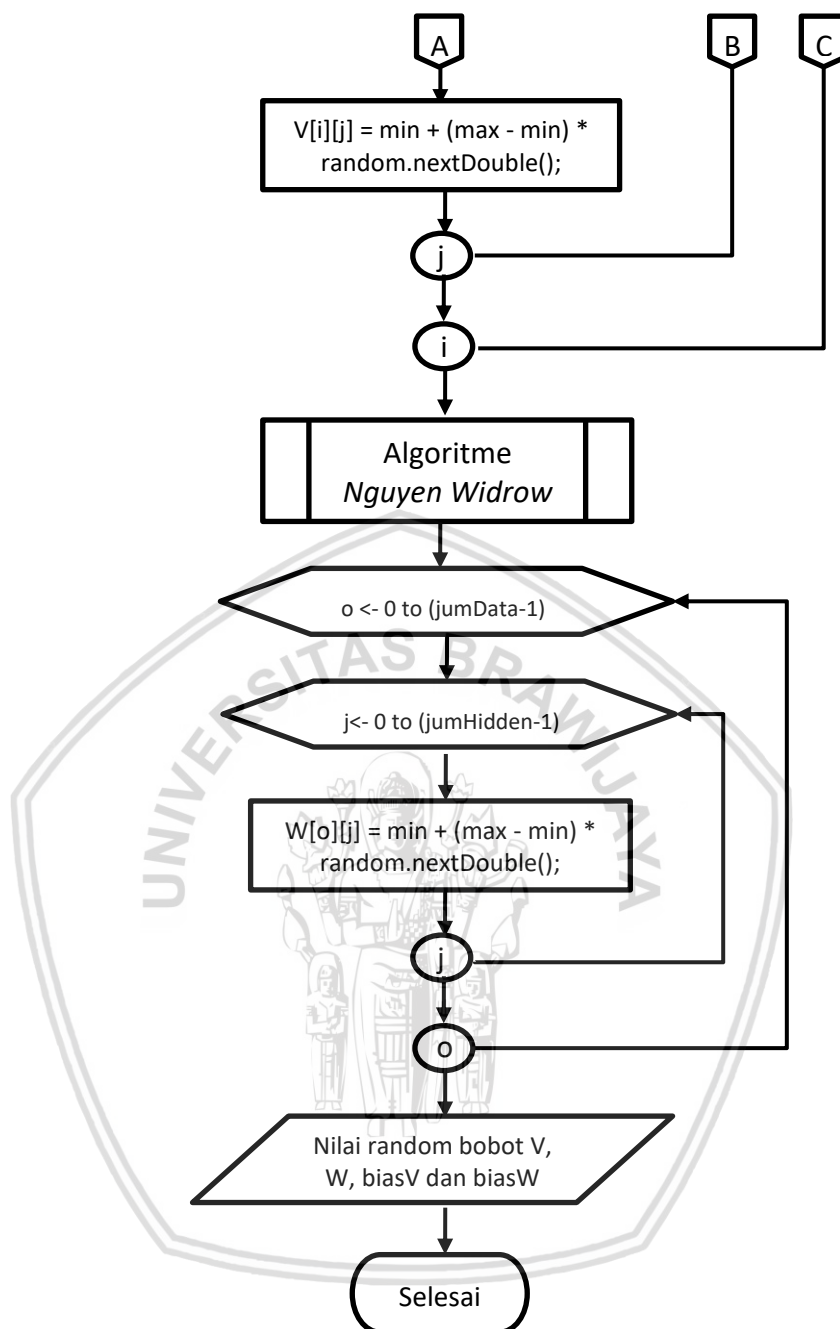


Gambar 4.3 flowchart Pelatihan Data

4.2.3 Inisialisasi Bobot Awal

inisialisasi bobot awal menggunakan algoritme *nguyen widrow*. Algoritme *nguyen widrow* merupakan algoritme dalam menentukan bobot yang terbaik untuk digunakan pada proses pelatihan data yang bertujuan mempercepat waktu komputasi. Flowchart inisialisasi bobot awal dapat dilihat pada gambar 4.4.

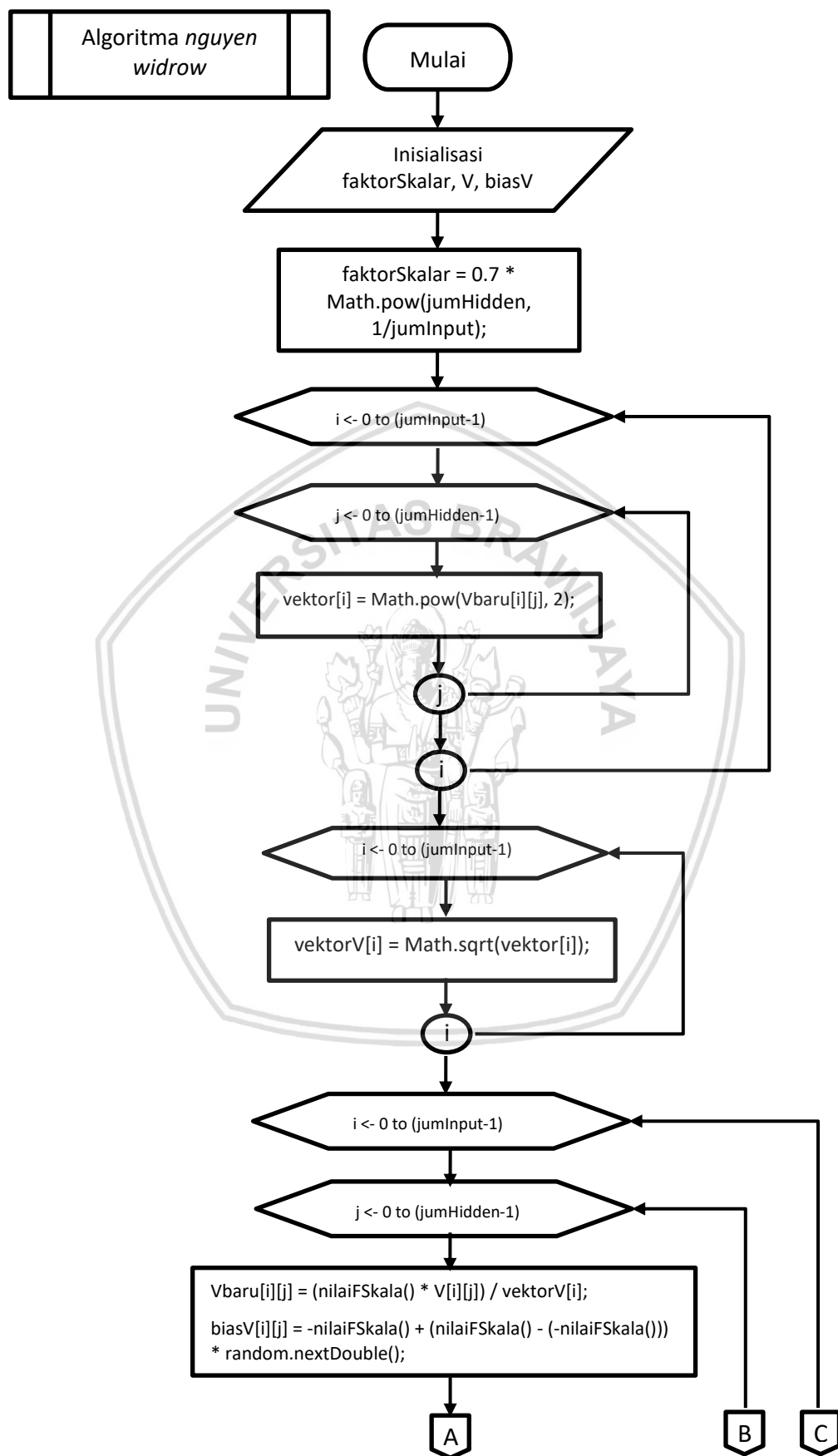


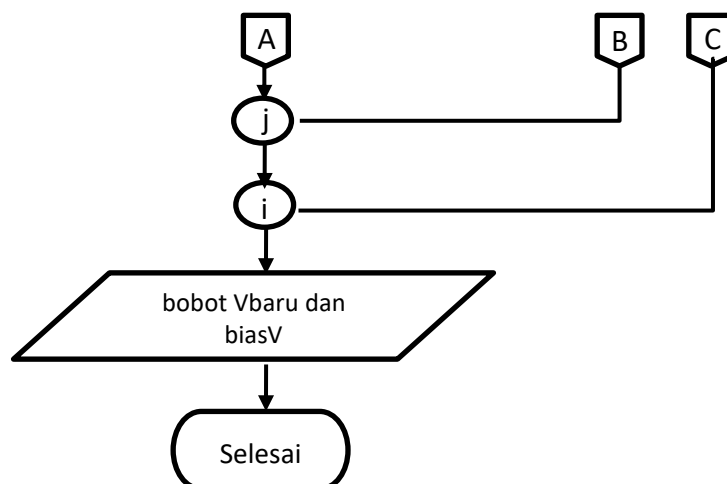


Gambar 4.4 Flowchart Inisialisasi Bobot Awal

4.2.3.1 Algoritme *nguyen widrow*

Pada proses inisialisasi bobot awal, diperlukan penentuan bobot awal yang terbaik agar dapat mempercepat fase pelatihan data, penentuan bobot V_{oj} dan V_{ij} menggunakan algoritme *nguyen widrow*, algoritme ini diharapkan dapat menentukan bobot awal terbaik yang akan digunakan pada fase pelatihan data. Flowchat algoritme *nguyen widrow* dapat dilihat pada gambar 4.5.

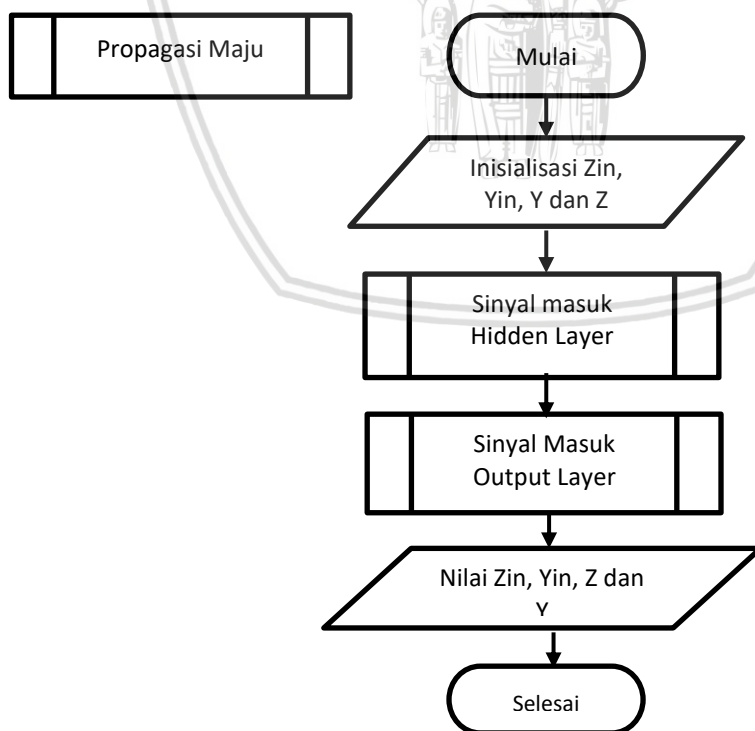




Gambar 4.5 Flowchart Algoritme *Nguyen Widrow*

4.2.4 Fase Propagasi Maju

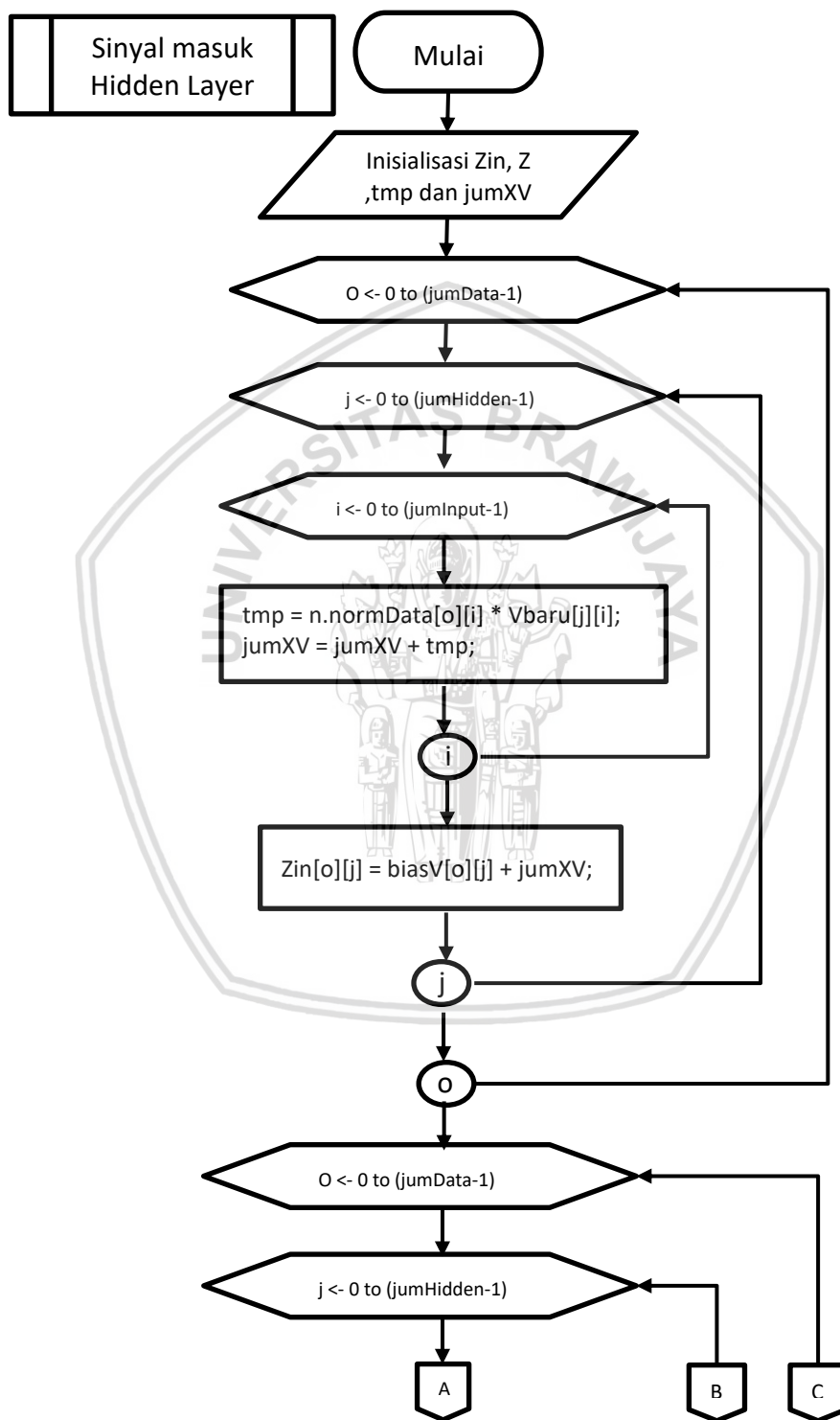
Pada fase ini nilai sinyal yang masuk ke *hidden* layer dan output layer akan di hitung. Kemudian nilai yang di hitung akan dilakukan aktivasi. Tahap-tahap perhitungan di fase propagasi maju terdapat 2 subproses yaitu Subproses sinyal masuk *hidden* layer, menjelaskan alur perhitungan sinyal yang masuk ke *hidden* layer (Z_{in}) dan perhitungan aktivasi Z_{in} (nilai Z). Subproses sinyal masuk output layer, menjelaskan alur perhitungan sinyal masuk output layer (Y_{in}) dan perhitungan aktivasi Y_{in} (nilai Y). Flowchart pada fase propagasi maju dapat dilihat pada gambar 4.6.

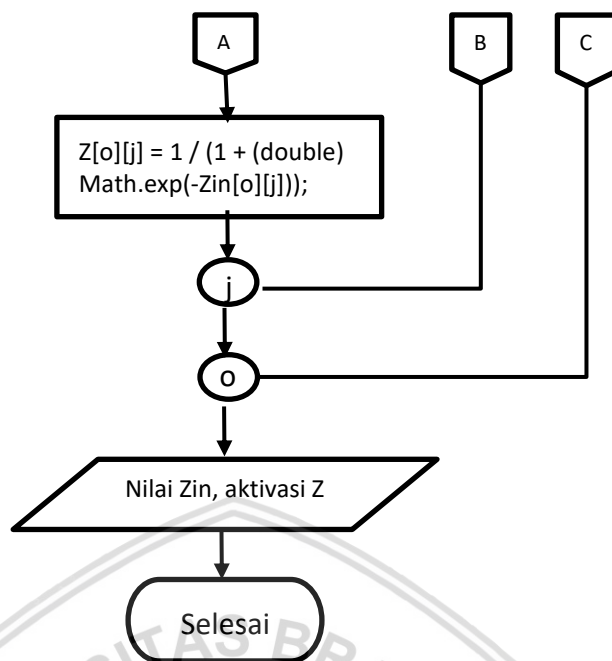


Gambar 4.6 flowchart Fase Propagasi Maju

4.2.4.1 Sinyal Masuk *Hidden Layer*

Pada subproses ini dilakukan perhitungan Z_{in} dan aktivasi Z_{in} (nilai Z) ketika sinyal masuk ke *hidden layer*. Flowchart alur kerjanya dapat dilihat pada gambar 4.7.

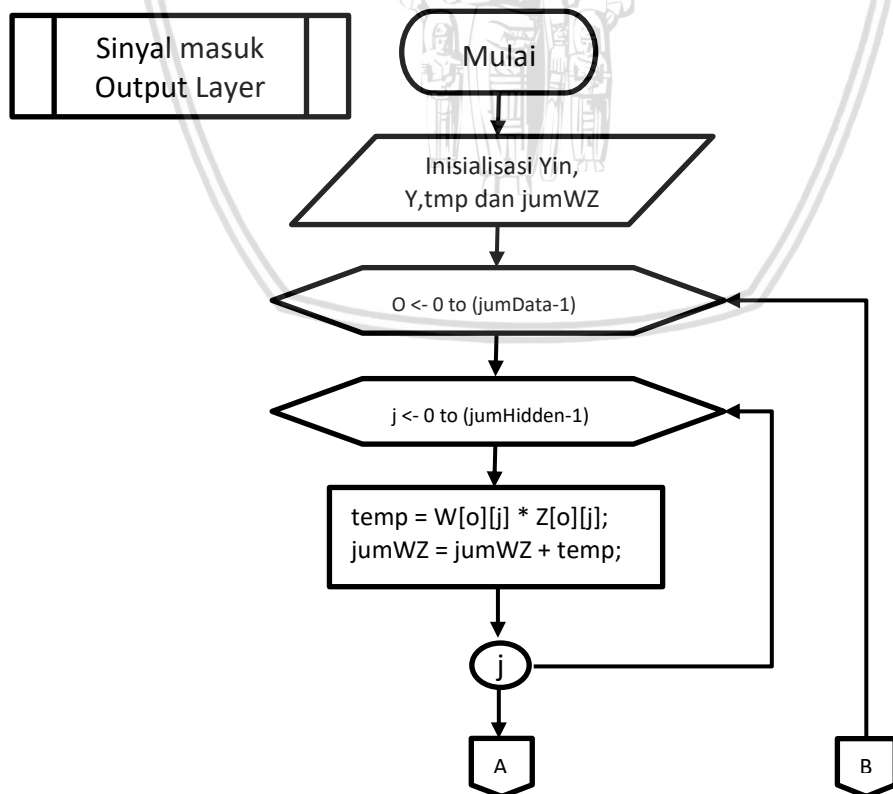


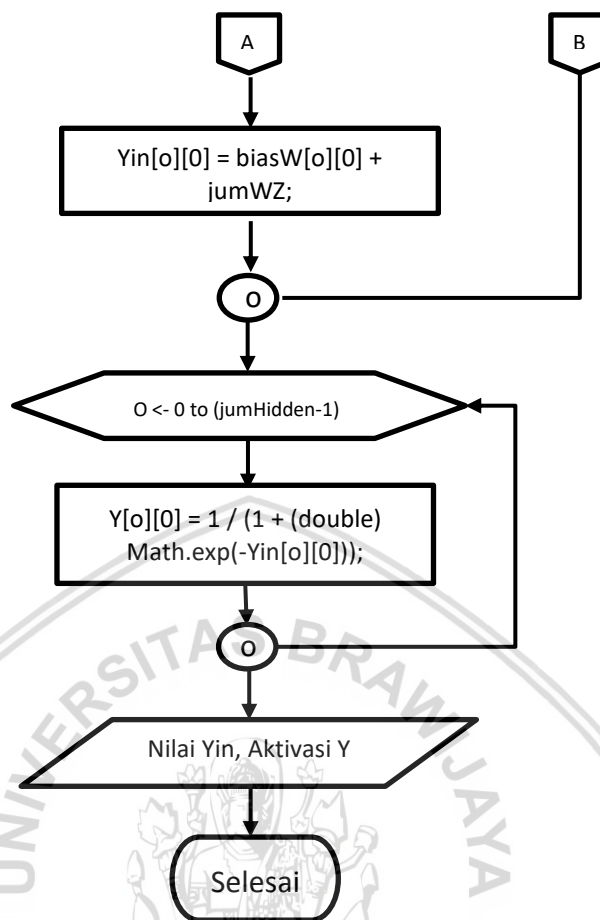


Gambar 4.7 Sinyal Masuk *Hidden Layer*

4.2.4.2 Sinyal Masuk Output layer

Subproses ini menghitung sinyal yang masuk ke output layer (Y_{in}) dan nilai aktifasinya Y_{in} (nilai Y). flowchart perhitungannya dapat dilihat pada gambar 4.8.

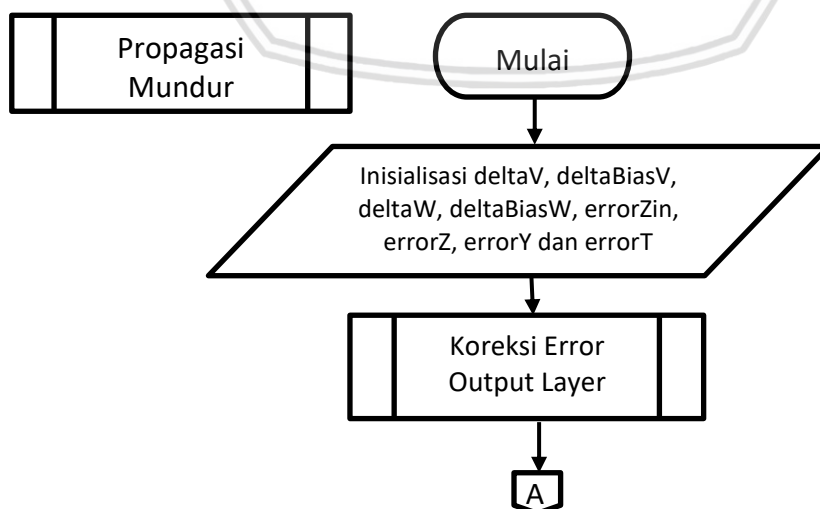


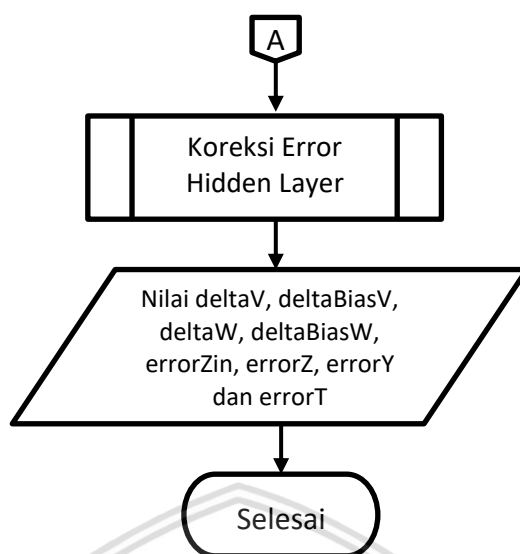


Gambar 4.8 Sinyal Masuk Output Layer

4.2.5 Fase Propagasi Mundur

Proses pelatihan data selanjutnya yaitu pada fase propagasi mundur. Pada fase ini nilai faktor koreksi w_{jk} , v_{ij} , w_{ok} , v_{ok} , Z_{in} , Z dan Y di hitung. Flowchart perhitungan faktor koreksi tersebut di tunjukkan pada gambar 4.9.

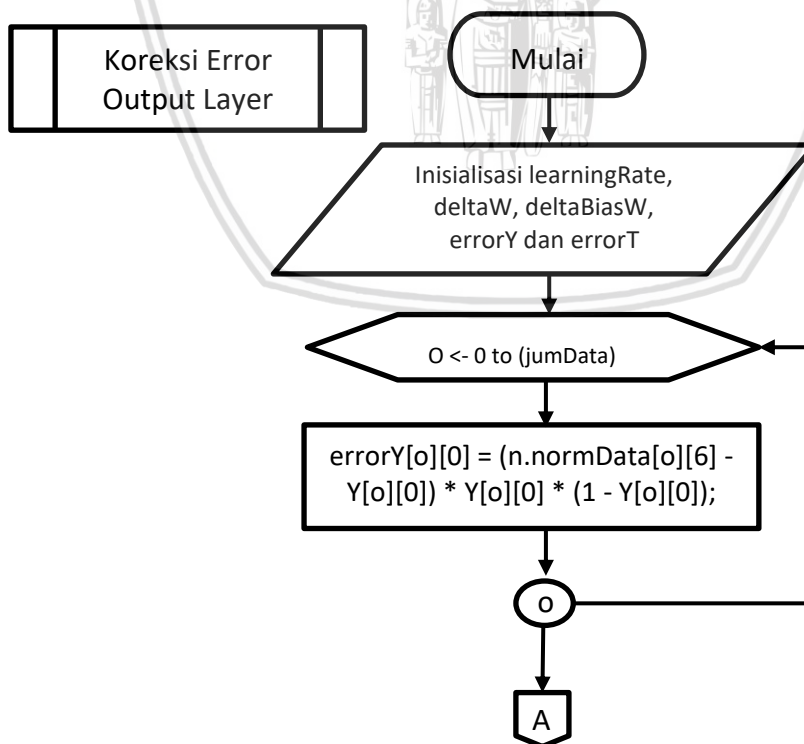


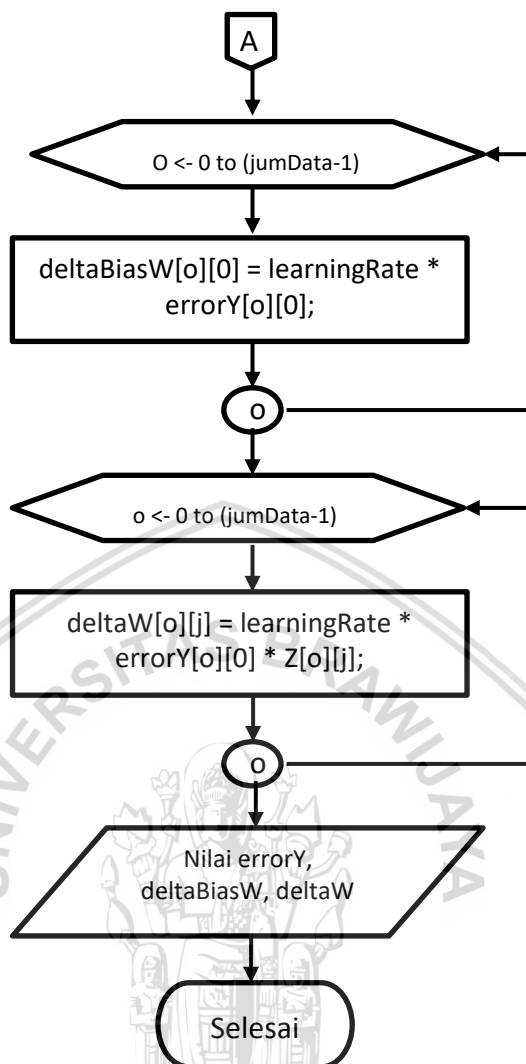


Gambar 4.9 Fase Propagasi Mundur

4.2.5.1 Koreksi Error Output Layer

Pada subproses ini berfokus pada perhitungan koreksi error yang menghubungkan *hidden* layer dengan output layer(Y). nilai yang di koreksi diantaranya nilai Wjk dan Wok yang nantinya digunakan untuk perhitungan pada data selajutnya atau iterasi selanjutnya. Flowchart perhitunganya dapat dilihat pada gambar 4.10.

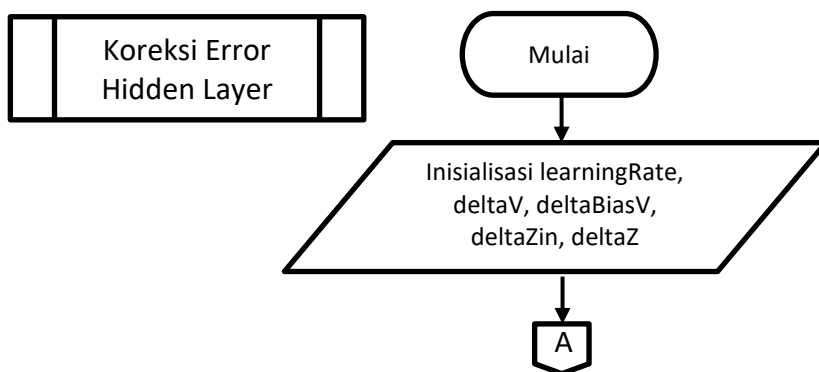


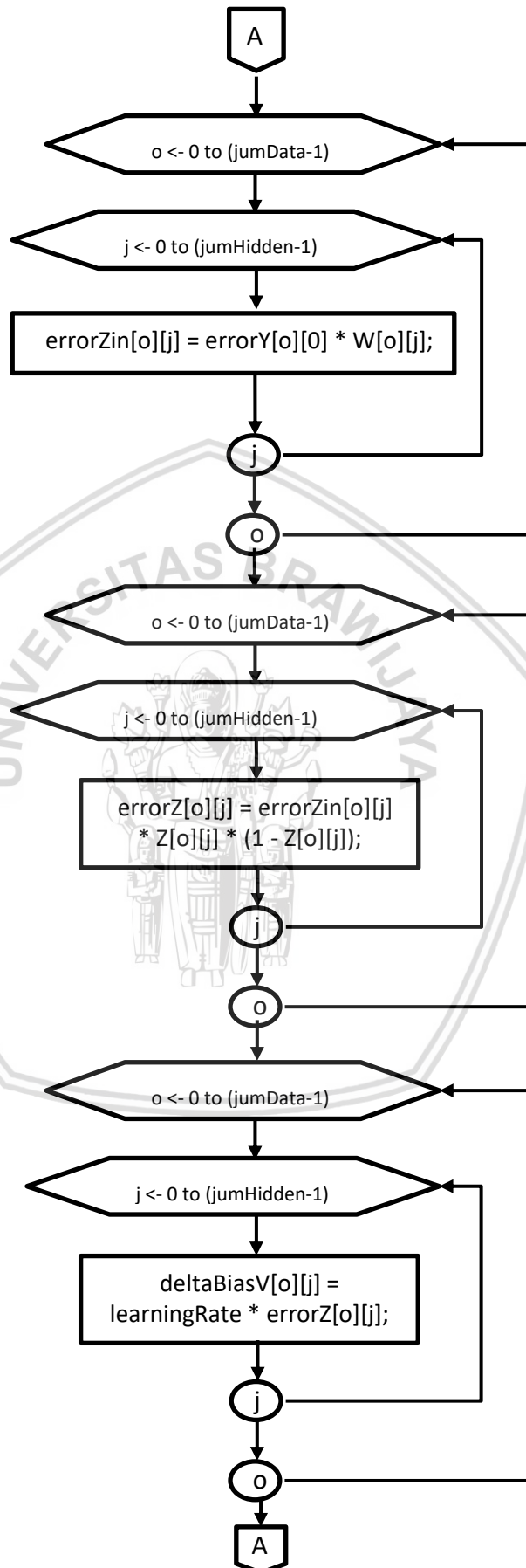


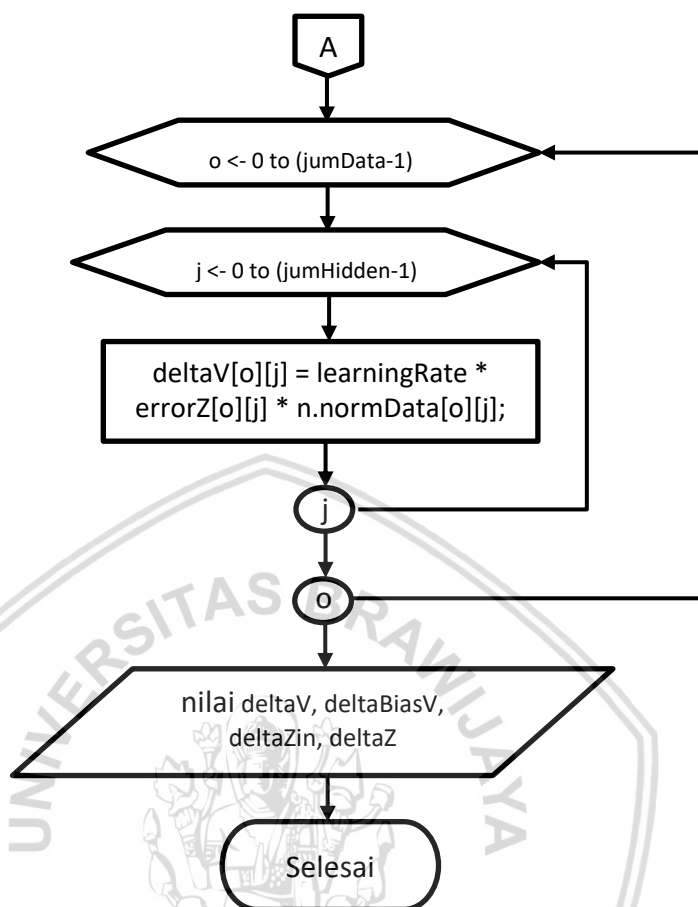
Gambar 4.10 Koreksi Error Output Layer

4.2.5.2 koreksi *Error Hidden Layer*

Setelah menghitung nilai koreksi error pada output layer subproses selanjutnya ialah perhitungan nilai koreksi error yang menghubungkan input layer dengan *hidden* layer di hitung. Nilai yang di koreksi meliputi bobot V_{ij} , bobot Bias V. Flowchart perhitungan nilai error dapat dilihat pada gambar 4.11.



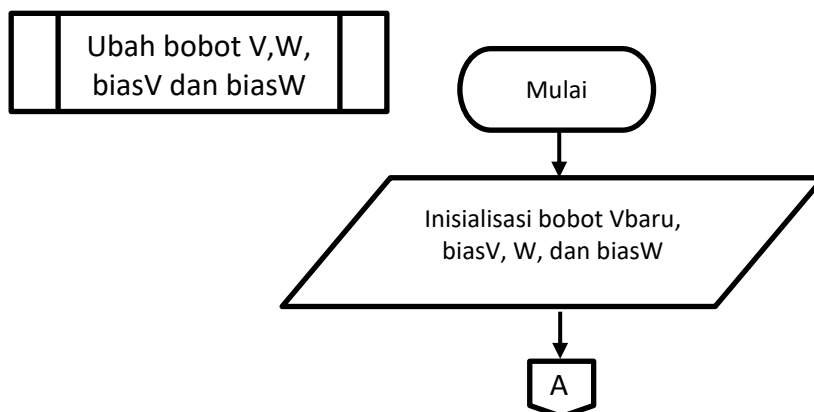


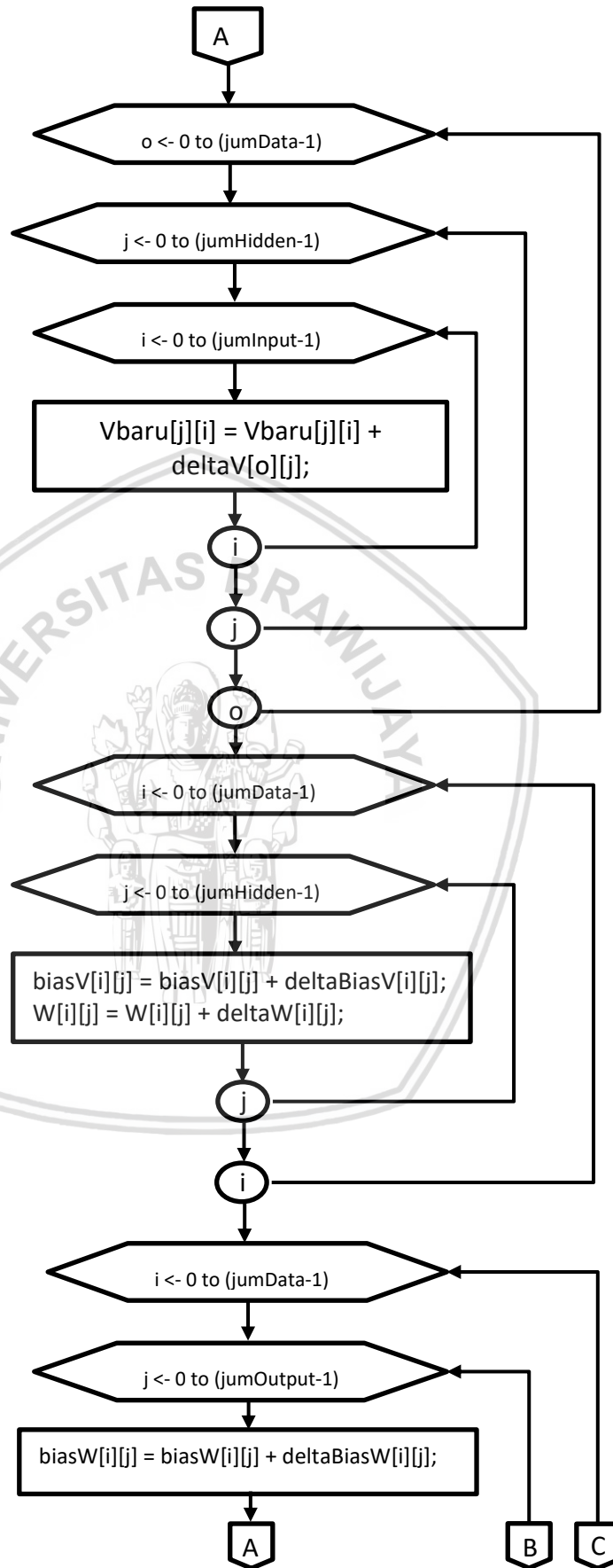


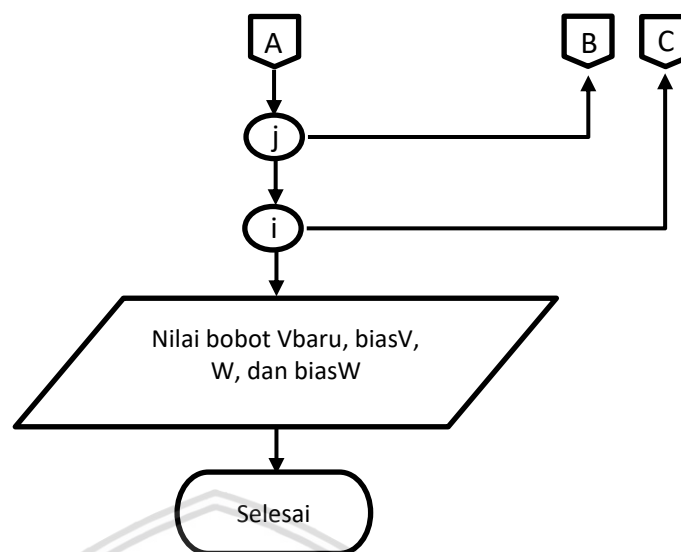
Gambar 4.11 Koreksi Error Hidden Layer

4.2.6 Ubah bobot w_{jk} , v_{ij} , w_{ok} , dan v_{oj}

Setelah melakukan pelatihan data pada fase propagasi maju kemudian propagasi mundur, selanjutnya ubah bobot w_{jk} , v_{ij} , w_{ok} , dan v_{oj} terhadap faktor koreksi yang telah di perhitungkan pada fase propagasi mundur. Flowchat ubah bobot w_{jk} , v_{ij} , w_{ok} , dan v_{ok} ditunjukkan pada gambar 4.12 berikut.







Gambar 4.12 flowchart ubah bobot w_{jk} , v_{ij} , w_{ok} , dan v_{ok}

4.3 Perhitungan Manualisasi

Pada bab ini menjelaskan tahap-tahap perhitungan *backpropagation* menggunakan data produksi dari tahun 2015-2017 pada PT. Sandabi Indah Lestari. Data Pelatihan yang digunakan pada proses manualisasi ini berjumlah 30 data dan data pengujian yang digunakan berjumlah 6 data. Pada proses pelatihan data nilai *learning rate* yang digunakan adalah 0.2, kemudian jumlah *neuron hidden layer* 2/3 dari jumlah neuron input layer di tambah dengan jumlah neuron output layer menghasilkan 5 *hidden layer*.

4.3.1 Normalisasi Data

Pada sub bab ini menjelaskan proses normalisasi data aktual yaitu data produksi mulai dari tahun 2015-2017 pada PT. sandabi indah lestari yang dapat dilihat pada tabel 4.2.

Tabel 4.2 Data produksi 2015 sampai 2017 PT. Sandabi Indah Lestari

No	X_1	X_2	X_3	X_4	X_5	X_6	T
1	16	2592	18	38	34	55	33
2	17	2592	18	34	32	40	29
3	15	33935	280	601	533	522	557
4	16	33935	280	438	488	590	487
5	14	21799	185	362	247	297	307
6	15	21799	185	240	245	286	288
7	13	87583	716	1151	1350	1241	1097
8	14	87583	716	1009	998	1174	1180
9	12	85510	700	1177	1222	1119	1110
10	13	85510	700	988	885	1138	1063
11	11	17970	149	289	244	268	252
12	12	17970	149	187	200	283	244

13	10	10583	92	136	157	134	124
14	11	10583	92	124	116	146	124
15	9	436	5	4	4	5	2
16	10	436	5	3	2	3	3
17	7	17352	157	215	193	142	140
18	8	17352	157	191	169	217	182
19	6	1339	15	13	13	12	13
20	7	1339	15	16	13	20	14
21	5	2575	13	10	13	10	9
22	6	2575	13	17	15	16	12
23	16	2592	18	34	55	33	35
24	17	2592	18	32	40	29	33
25	15	33935	280	533	522	557	469
26	16	33935	280	488	590	487	440
27	14	21799	185	247	297	307	255
28	15	21799	185	245	286	288	248
29	13	87583	716	1350	1241	1097	1149
30	14	87583	716	998	1174	1180	979

Perhitungan normalisasi data menggunakan 30 data dari total jumlah data latih yaitu 260 data. Normalisasi data dilakukan dengan cara menggunakan persamaan 2.20. Pertama cari nilai tertinggi dan terendah pada setiap parameter. Jumlah parameter pada penelitian ini berjumlah 6 parameter input dan 1 parameter target. Pada Tabel 4.3 dibawah ini merupakan nilai terendah dan tertinggi atau nilai minimal dan maksimal dari masing-masing parameter.

Tabel 4.3 Data terendah dan tertinggi tiap parameter

	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	T
Min	4	436	5	2	2	2	2
Max	17	87583	715.92	1355	1355	1417	1417

Kemudian hitung nilai normalisasinya menggunakan persamaan 2.17. Contoh perhitungan pada data pertama:

$$x_1 = \frac{16 - 4}{17 - 4} (0.8) + 0.1 = 0.838462$$

$$x_2 = \frac{2592 - 4}{87583 - 436} (0.8) + 0.1 = 0.119792$$

$$x_3 = \frac{18 - 5}{716 - 5} (0.8) + 0.1 = 0.114990371$$

$$x_4 = \frac{38 - 2}{1355 - 2} (0.8) + 0.1 = 0.121286$$

$$x_5 = \frac{34 - 2}{1355 - 2} (0.8) + 0.1 = 0.118921$$

$$x_6 = \frac{55 - 2}{1417 - 2} (0.8) + 0.1 = 0.129965$$

$$t_1 = \frac{33 - 2}{1417 - 2} (0.8) + 0.1 = 0.117527$$

Tabel 4.4 merupakan hasil normalisasi dari 30 data produksi 2015 sampai 2017 PT. Sandabi Indah Lestari.

Tabel 4.4 Data Produksi setelah di normalisasi

No	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	T
1	0,838462	0,119792	0,11499	0,121286	0,118921	0,129965	0,117527
2	0,9	0,119792	0,11499	0,118921	0,117738	0,121484	0,115265
3	0,776923	0,407517	0,40994	0,454176	0,413969	0,393993	0,413781
4	0,838462	0,407517	0,40994	0,357797	0,387361	0,432438	0,374205
5	0,715385	0,29611	0,303343	0,31286	0,244863	0,266784	0,272438
6	0,776923	0,29611	0,303343	0,240724	0,243681	0,260565	0,261696
7	0,653846	0,9	0,9	0,779379	0,897044	0,800495	0,719081
8	0,715385	0,9	0,9	0,695418	0,688914	0,762615	0,766007
9	0,592308	0,88097	0,881895	0,794752	0,82136	0,731519	0,726431
10	0,653846	0,88097	0,881895	0,683001	0,622099	0,742261	0,699859
11	0,530769	0,26096	0,262926	0,269697	0,243089	0,250389	0,241343
12	0,592308	0,26096	0,262926	0,209387	0,217073	0,258869	0,23682
13	0,469231	0,193148	0,198152	0,179231	0,191648	0,174629	0,168975
14	0,530769	0,193148	0,198152	0,172136	0,167406	0,181413	0,168975
15	0,407692	0,1	0,1	0,101183	0,101183	0,101696	0,1
16	0,469231	0,1	0,1	0,100591	0,1	0,100565	0,100565
17	0,284615	0,255287	0,271282	0,225942	0,212934	0,179152	0,178021
18	0,346154	0,255287	0,271282	0,211752	0,198744	0,221555	0,201767
19	0,223077	0,108289	0,111808	0,106504	0,106504	0,105654	0,106219
20	0,284615	0,108289	0,111808	0,108278	0,106504	0,110177	0,106784
21	0,161538	0,119636	0,10945	0,10473	0,106504	0,104523	0,103958
22	0,223077	0,119636	0,10945	0,108869	0,107687	0,107915	0,105654
23	0,838462	0,119792	0,11499	0,118921	0,131338	0,117527	0,118657
24	0,9	0,119792	0,11499	0,117738	0,122469	0,115265	0,117527
25	0,776923	0,407517	0,40994	0,413969	0,407465	0,413781	0,364028
26	0,838462	0,407517	0,40994	0,387361	0,447672	0,374205	0,347633
27	0,715385	0,29611	0,303343	0,244863	0,274427	0,272438	0,243039
28	0,776923	0,29611	0,303343	0,243681	0,267923	0,261696	0,239081
29	0,653846	0,9	0,9	0,897044	0,832594	0,719081	0,748481
30	0,715385	0,9	0,9	0,688914	0,792979	0,766007	0,652367

4.3.2 Inisialisasi Bobot Awal

Pada tahap inisialisasi bobot awal digunakan algoritme *nguyen widrow*. Algoritme *nguyen widrow* membantu untuk mempercepat waktu pelatihan karena algoritme ini mampu menemukan bobot yang terbaik untuk digunakan pada inisialisasi awal. Berikut langkah-langkah algoritme *nguyen widrow* akan di jabarkan. Pada penelitian ini arsitektur jaringan yang digunakan adalah 6-5-1 yaitu 6 neuron input layer, 5 *neuron hidden layer* dan 1 neuron output layer.

1. inisialisasi v_{ij} (bobot antara nilai neuron input dan neuron *hidden*) secara acak dengan range antara -0.5 sampai 0.5, hasil inisialisasi dapat dilihat pada tabel 4.5.

Tabel 4.5 Bobot v_{ij} random antara -0.5 sampai 0.5

J	Vij					
	1	2	3	4	5	6
1	0.2	-0.45	-0.31	0.35	0.42	0.14
2	-0.32	0.21	0.1	0.13	-0.4	-0.18
3	0.12	0.33	-0.28	0.47	0.34	-0.11
4	0.45	0.26	0.16	-0.12	-0.46	0.19
5	0.22	0.35	0.1	0.2	0.16	0.37

2. Selanjutnya hitung nilai β (faktor skala) menggunakan persamaan 2.18.

$$\beta = 0.7^n \sqrt{p}$$

Keterangan :

n = jumlah unit input

p = jumlah unit *hidden*

$$\beta = 0.7^6 \sqrt{5} = 0.92$$

3. Hasil perhitungan nilai β digunakan sebagai range bias, maka range bias antara -0.92 sampai 0.92. kemudian hitung nilai $||v_{ij}||$ (panjang vector pada setiap *hidden layer*). Menggunakan persamaan 2.17.

$$||v_{ij}|| = \sqrt{v_{1j}^2 + v_{2j}^2 + \dots + v_{nj}^2}$$

Berikut perhitungan $||v_1||$ sampai $||v_5||$:

$$||v_1|| =$$

$$\sqrt{(0.2)^2 + (-0.45)^2 + (0.31)^2 + (0.35)^2 + (0.42)^2 + (0.14)^2} = 0.810617$$

$$||v_2|| =$$

$$\sqrt{(-0.32)^2 + (0.21)^2 + (0.1)^2 + (0.13)^2 + (-0.4)^2 + (-0.18)^2} = 0.604814$$

$$||v_3|| =$$

$$\sqrt{(0.12)^2 + (0.33)^2 + (-0.28)^2 + (0.47)^2 + (0.34)^2 + (-0.11)^2} = 0.741822$$

$$||v_4|| =$$

$$\sqrt{(0.45)^2 + (0.26)^2 + (0.16)^2 + (-0.12)^2 + (-0.46)^2 + (0.19)^2} = 0.74686$$

$$||v_5|| =$$

$$\sqrt{(0.22)^2 + (-0.35)^2 + (0.1)^2 + (0.2)^2 + (0.16)^2 + (0.37)^2} = 0.619193$$

4. Kemudian menghitung bobot v_{ij} yang akan digunakan pada inisialisasi awal dari input layer ke *hidden layer* menggunakan persamaan 2.19. berikut contoh data pertama.

$$v_{ij} = \frac{\beta v_{ij}(\text{lama})}{||v_{ij}||}$$

$$v_{11} = \frac{0.92 \times 0.2}{0.81061} = 0.225843$$

Berikut tabel 4.6 perhitungan bobot v_{ij} baru menggunakan persamaan 2.19 yang nantinya digunakan sebagai inisialisasi awal dari neuron input layer ke *neuron hidden layer*.

Tabel 4.6 Bobot v_{ij} baru yang digunakan sebagai inisialisasi awal

J	Vij					
	1	2	3	4	5	6
1	0.225843	-0.508148	-0.350057	0.395226	0.474271	0.158090
2	-0.484307	0.317827	0.151346	0.196750	-0.605384	-0.272423
3	0.148073	0.407199	-0.345503	0.579951	0.419539	-0.135733
4	0.551526	0.318660	0.196098	-0.147074	-0.563783	0.232867
5	0.325229	0.517410	0.147832	0.295663	0.236530	0.546977

5. Kemudian tentukan secara acak bias awal yang akan digunakan berdasarkan nilai β yang di dapat dari perhitungan sebelumnya yang menghasilkan range batasan antara -0.92 sampai 0.92. penentuan acak bias sesuai range dapat dilihat pada tabel 4.7.

Tabel 4.7 Bias awal yang ditentukan acak antara -0.92 sampai 0.92

V	J				
	1	2	3	4	5
0	0.88	0.65	-0.61	-0.34	0.54

6. Langkah terakhir pada proses *nguyen widrow*, tentukan bobot w_{jk} dan bobot bias w_{ok} dari *hidden* layer ke output layer dengan range antara -0.5 sampai 0.5. Hasil penentuan dapat dilihat pada tabel 4.8 dan 4.9.

Tabel 4.8 Bobot w_{ok} random antara -0.5 sampai 0.5

	0
w_{ok}	0.43

Tabel 4.9 Bobot w_{jk} random antara -0.5 sampai 0.5

J	1	2	3	4	5
w_{jk}	0.22	-0.12	-0.33	0.21	0.12

4.3.3 Fase Propagasi Maju

Tahap ini dilakukan setelah melakukan normalisasi data dan inisialisasi awal. Pada fase propagasi maju perhitungan di fokuskan pada sinyal yang masuk ke *hidden* layer (z_{in_j}) dan sinyal yang masuk ke output layer (y_{in_j}).

1. Pertama hitung sinyal yang masuk ke *hidden* layer menggunakan persamaan 2.1:

$$z_{in_j} = v_{0j} + \sum_{k=0}^n x_i v_{ij}$$

$$z_{in_1} = 0.88 + (0.838461538 \times 0.225843) + (0.119792 \times -0.508148) + (0.114990371 \times -0.350057) + (0.121286031 \times 0.395226) + (0.118921 \times 0.474271) + (0.129964664 \times 0.158090) = 1.093118$$

$$z_{in_2} = 0.65 + (0.838461538 \times -0.484307) + (0.119792 \times 0.317827) + (0.114990371 \times 0.151346) + (0.121286031 \times 0.196750) + (0.118921 \times -0.605384) + (0.129964664 \times -0.272423) = 0.215868$$

$$z_{in_3} = -0.61 + (0.838461538 \times 0.148073) + (0.119792 \times 0.407199) + (0.114990371 \times -0.345503) + (0.121286031 \times 0.579951) + (0.118921 \times 0.419539) + (0.129964664 \times -0.135733) = -0.374206$$

$$z_{in_4} = -0.34 + (0.838461538 \times 0.551526) + (0.119792 \times 0.318660) + (0.114990371 \times 0.196098) + (0.121286031 \times -0.147074) + (0.118921 \times -0.563783) + (0.129964664 \times 0.232867) = 0.177606$$

$$z_{in_5} = 0.54 + (0.838461538 \times 0.325229) + (0.119792 \times 0.517410) + (0.114990371 \times 0.147832) + (0.121286031 \times 0.295663) + (0.118921 \times 0.236530) + (0.129964664 \times 0.546977) = 1.026749$$

Setelah menghitung nilai z_{in_j} kemudian hitung aktivasinya menggunakan persamaan 2.2:

$$z_j = f(z_{in_j}) = \frac{1}{1 + \exp^{-z_{in_j}}}$$

$$z_1 = f(z_{in_2}) = \frac{1}{1 + \exp^{-1.093118}} = 0.748968$$

$$z_2 = f(z_{in_2}) = \frac{1}{1 + \exp^{-0.215868}} = 0.553758$$

$$z_3 = f(z_{in_3}) = \frac{1}{1 + \exp^{0.374206}} = 0.407525$$

$$z_4 = f(z_{in_4}) = \frac{1}{1 + \exp^{-0.177606}} = 0.544285$$

$$z_5 = f(z_{in_5}) = \frac{1}{1 + \exp^{-1.026749}} = 0.736285$$

2. Hitung sinyal yang masuk ke layer output menggunakan persamaan 2.3:

$$y_{in_k} = w_{0k} + \sum_{j=0}^p z_j w_{jk}$$

$$y_{in_k} = w_{0k} + z_1 w_{1k} + z_2 w_{2k} + z_3 w_{3k} + z_4 w_{4k} + z_5 w_{5k}$$

$$y_{in_1} = 0.43 + (0.748968 \times 0.22) + (0.553758 \times -0.12) + (0.407525 \times -0.33) + (0.544285 \times 0.21) + (0.736285 \times 0.12) = 0.596493$$

Setelah menghitung y_{in_k} , kemudian hitung aktivasinya menggunakan persamaan 2.4:

$$y_k = f(y_{in_k}) = \frac{1}{1 + \exp^{-y_{in_k}}}$$

$$y_k = f(y_{in_k}) = \frac{1}{1 + \exp^{-0.596493}} = 0.644854$$

4.3.4 Fase Propagasi Mundur

Fase Propagasi Mundur berfokus pada perhitungan faktor koreksi bobot w_{jk} dan v_{ij} berdasarkan error output dari y_k , tahap pertama hitung nilai δ_k (faktor koreksi w_{jk}) menggunakan persamaan 2.5:

$$\delta_k = (t_k - y_k)y_k(1 - y_k)$$

$$\delta_1 = (0.117527 - 0.644854)0.639233(1 - 0.644854) = -0.120767$$

Setelah menghitung faktor koreksi bobot w_{jk} , hitung Δw_{jk} yang berguna mengubah bobot w_{jk} dengan *learning rate* 0.2. untuk menghitung Δw_{jk} menggunakan persamaan 2.6:

$$\Delta w_{jk} = \alpha \delta_k z_j$$

$$\Delta w_{11} = 0.2 \times -0.120767 \times 0.72626 = -0.018090$$

$$\Delta w_{12} = 0.2 \times -0.120767 \times 0.613318 = -0.013375$$

$$\Delta w_{13} = 0.2 \times -0.120767 \times 0.385685 = -0.009843$$

$$\Delta w_{14} = 0.2 \times -0.120767 \times 0.474413 = -0.013146$$

$$\Delta w_{15} = 0.2 \times -0.120767 \times 0.695874 = -0.017784$$

Selanjutnya menghitung Δw_{ok} , yang berguna untuk mengubah bobot w_{ok} untuk data selanjutnya. Gunakan persamaan 2.7:

$$\Delta w_{ok} = \alpha \delta_k$$

$$\Delta w_{o1} = 0.2 \times -0.120767 = -0.024153$$

Kemudian menghitung delta bobot *hidden layer* (δ_{in_j}) yang nantinya digunakan dalam perhitungan faktor koreksi error bobot *hidden layer*. Gunakan persamaan 2.8:

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk}$$

$$\delta_{in_1} = -0.120767 \times 0.22 = -0.026569$$

$$\delta_{in_2} = -0.120767 \times -0.12 = 0.014492$$

$$\delta_{in_3} = -0.120767 \times -0.33 = 0.039853$$

$$\delta_{in_4} = -0.120767 \times 0.21 = -0.025361$$

$$\delta_{in_5} = -0.120767 \times 0.12 = -0.014492$$

Setelah itu hitung faktor koreksi error bobot *hidden layer* menggunakan persamaan 2.9:

$$\delta_j = \delta_{in_j} z_j (1 - z_j)$$

$$\delta_1 = -0.026569 \times 0.748968 (1 - 0.748968) = -0.004995$$

$$\delta_2 = 0.014492 \times 0.553758 (1 - 0.553758) = 0.003581$$

$$\delta_3 = 0.039853 \times 0.407525 (1 - 0.407525) = 0.009622$$

$$\delta_4 = -0.025361 \times 0.544285 (1 - 0.544285) = -0.006291$$

$$\delta_5 = -0.014492 \times 0.736285(1 - 0.736285) = -0.002814$$

Hitung koreksi error bobot $v_{ij}(\Delta v_{ij})$ menggunakan persamaan 2.10:

$$\Delta v_{ij} = \alpha \delta_j x_i$$

$$\Delta v_{11} = 0.2 \times -0.004995 \times 0.346154 = -0.000838$$

Lakukan perhitungan hingga Δv_{65} . Berikut tabel perhitungan yang telah dilakukan dapat dilihat pada tabel 4.10.

Tabel 4.10 Δv_{ij} dari data ke 1

J	Δv_{ij}					
	1	2	3	4	5	6
1	-0.000838	-0.000120	-0.000115	-0.000121	-0.000119	-0.000130
2	0.000601	0.000086	0.000082	0.000087	0.000085	0.000093
3	0.001614	0.000231	0.000221	0.000233	0.000229	0.000250
4	-0.001055	-0.000151	-0.000145	-0.000153	-0.000150	-0.000164
5	-0.000472	-0.000067	-0.000065	-0.000068	-0.000067	-0.000073

4.3.5 Ubah bobot w_{jk} , v_{ij} , w_{ok} , dan v_{oj}

Tahap terakhir pada proses pelatihan *backpropagation* ialah merubah bobot w_{jk} dan v_{ij} yang nantinya digunakan untuk data selanjutnya agar hasil output yaitu y_k mendekati nilai t_k (target pada data ke - k).

1. Pertama langkah-langkah untuk mengubah bobot v_{ij} yang menghubungkan input layer dengan *hidden* layer. Setelah menghitung Δv_{ij} kemudian hitung v_{ij} baru menggunakan persamaan di 2.13:

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij}$$

Tabel 4.11 hasil dari perhitungan $v_{ij}(\text{baru})$ yang akan digunakan pada perhitungan data selanjutnya

Tabel 4.11 bobot v_{ij} baru untuk data selanjutnya

J	v_{ij}					
	1	2	3	4	5	6
1	0.225006	-0.508267	-0.350172	0.395105	0.474152	0.157961
2	-0.483707	0.317913	0.151428	0.196837	-0.605299	-0.272330
3	0.149686	0.407430	-0.345281	0.580184	0.419768	-0.135483
4	0.550472	0.318509	0.195954	-0.147226	-0.563932	0.232703
5	0.324757	0.517343	0.147767	0.295595	0.236463	0.546903

Kemudian hitung Δv_{oj} untuk merubah bobot inisialisasi v_{oj} yang akan digunakan pada data selanjutnya menggunakan persamaan 2.11:

$$\Delta v_{oj} = \alpha \delta_j$$

$$\Delta v_{o1} = 0.2 \times -0.004995 = -0.000999$$

$$\Delta v_{o2} = 0.2 \times 0.003581 = 0.0007162$$

$$\Delta v_{o3} = 0.2 \times 0.009622 = 0.0019244$$

$$\Delta v_{04} = 0.2 \times -0.006291 = -0.0012582$$

$$\Delta v_{05} = 0.2 \times -0.002814 = -0.0005628$$

Setelah menghitung Δv_{oj} , kita dapat memperbarui bobot v_{oj} dengan menggunakan persamaan 2.14:

$$v_{oj}(\text{baru}) = v_{oj}(\text{lama}) + \Delta v_{oj}$$

$$v_{01}(\text{baru}) = 0.88 + -0.000999 = 0.879001$$

$$v_{02}(\text{baru}) = 0.65 + 0.0007162 = 0.650716$$

$$v_{03}(\text{baru}) = -0.61 + 0.0019244 = -0.608076$$

$$v_{04}(\text{baru}) = -0.34 + -0.0012582 = -0.341258$$

$$v_{05}(\text{baru}) = 0.54 + -0.0005628 = 0.539437$$

2. Kemudian tahap mengubah bobot w_{jk} dan w_{ok} . Gunakan persamaan 2.12 untuk mengubah bobot w_{jk} untuk perhitungan data selanjutnya.

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk}$$

$$w_{11}(\text{baru}) = 0.22 + -0.018090 = 0.201910$$

$$w_{12}(\text{baru}) = -0.12 + -0.013375 = -0.133375$$

$$w_{13}(\text{baru}) = -0.33 + -0.009843 = -0.339843$$

$$w_{14}(\text{baru}) = 0.21 + -0.013146 = 0.196854$$

$$w_{15}(\text{baru}) = 0.12 + -0.017784 = 0.102216$$

hitung bobot w_{ok} menggunakan persamaan 2.15:

$$w_{ok}(\text{baru}) = w_{ok}(\text{lama}) + \Delta w_{jk}$$

$$w_{15}(\text{baru}) = 0.43 + -0.024153 = 0.405847$$

Setelah perhitungan di atas didapatkan bobot w_{jk} , v_{ij} , w_{ok} , dan v_{ok} yang baru. Perhitungan dari fase propagasi maju, fase propagasi mundur, dan ubah bobot menggunakan data latih ke 1, perhitungan dilakukan secara terus menerus hingga data latih ke 30. Pada proses Manualisasi yang penulis lakukan hanya sampai 1 iterasi saja. Berikut tabel 4.12 hingga tabel 4.15 merupakan bobot v_{ij} , v_{ok} , w_{jk} dan w_{ok} pada data ke 30 serta tabel 4.16 nilai target(t) dan nilai output (y_k) mulai dari data latih ke-1 hingga data ke-30.

Tabel 4.12 Bobot v_{ij} pada data latih ke-30

j	v_{ij}					
	1	2	3	4	5	6
1	0.221666	-0.509005	-0.350928	0.394315	0.473433	0.157204
2	-0.476487	0.318136	0.151681	0.197201	-0.604980	-0.271810
3	0.163592	0.408476	-0.344173	0.581488	0.420936	-0.133981
4	0.545293	0.317646	0.195066	-0.148173	-0.564792	0.231759
5	0.324417	0.517102	0.147521	0.295346	0.236247	0.546684

Tabel 4.13 Bobot v_{oj} pada data latih ke-30

v_{oj}	J				
	1	2	3	4	5
30	0.873208	0.666236	-0.579337	-0.350839	0.539567

Tabel 4.14 Bobot w_{jk} pada data latih ke-30

w_{jk}	J				
	1	2	3	4	5
30	-0.018264	-0.312366	-0.457487	0.050480	-0.111091

Tabel 4.15 Bobot w_{ok} pada data latih ke-30

w_{ok}	K
	1
30	0.105001

Tabel 4.16 Perbandingan nilai target (T) dan output (Y)

Data ke	T	Y
1	0.117527	0.644854
2	0.115265	0.628717
3	0.413781	0.607755
4	0.374205	0.603978
5	0.272438	0.594086
6	0.261696	0.584323
7	0.719081	0.562665
8	0.766007	0.570160
9	0.726431	0.575808
10	0.699859	0.584783
11	0.241343	0.593169
12	0.236820	0.582704
13	0.168975	0.569702
14	0.168975	0.556757
15	0.100000	0.543375
16	0.100565	0.528658
17	0.178021	0.509427
18	0.201767	0.498738
19	0.106219	0.489908
20	0.106784	0.477101
21	0.103958	0.463511
22	0.105654	0.452328
23	0.118657	0.442889
24	0.117527	0.431839
25	0.364028	0.411590
26	0.347633	0.410301
27	0.243039	0.411861

28	0.239081	0.406202
29	0.748481	0.380203
30	0.652367	0.399308

Kemudian hitung tingkat keakuratan dari hasil pelatihan data menggunakan rumus MSE (Mean Square Error). perhitungan MSE menggunakan semua data dari data ke 1 hingga data ke 30. Perhitungan menggunakan persamaan 2.16:

$$MSE = \sum_{t=1}^{30} \frac{(t - y)^2}{30}$$

$$MSE = 0.102356$$

4.3.6 Pengujian data

Setelah melakukan proses pelatihan data mulai dari fase propagasi maju, propagasi mundur dan merubah bobot, maka tahap selanjutnya melakukan pengujian dari hasil pelatihan. Pengujian menggunakan 6 data uji yang telah di normalisasi pada tabel 4.17:

Tabel 4.17 data uji yang sudah di normalisasi

No	X1	X2	X3	X4	X5	X6	T
1	0.592308	0.880970	0.881895	0.821360	0.760458	0.726431	0.681201
2	0.653846	0.880970	0.881895	0.622099	0.771693	0.699859	0.637668
3	0.530769	0.260960	0.262926	0.243089	0.257280	0.241343	0.237385
4	0.592308	0.260960	0.262926	0.217073	0.266149	0.236820	0.232297
5	0.469231	0.193148	0.198152	0.191648	0.178049	0.168975	0.167845
6	0.592308	0.880970	0.881895	0.821360	0.760458	0.726431	0.681201

Berikut tabel 4.18 hingga tabel 4.21 merupakan bobot w_{jk} , v_{ij} , w_{ok} , dan v_{oj} yang digunakan pada proses pengujian.

Tabel 4.18 Bobot v_{ij}

j	v_{ij}					
	1	2	3	4	5	6
1	0.221666	-0.509005	-0.350928	0.394315	0.473433	0.157204
2	-0.476487	0.318136	0.151681	0.197201	-0.604980	-0.271810
3	0.163592	0.408476	-0.344173	0.581488	0.420936	-0.133981
4	0.545293	0.317646	0.195066	-0.148173	-0.564792	0.231759
5	0.324417	0.517102	0.147521	0.295346	0.236247	0.546684

Tabel 4.19 Bobot v_{oj}

v_{oj}	J				
	1	2	3	4	5
30	0.873208	0.666236	-0.579337	-0.350839	0.539567

Tabel 4.20 Bobot w_{jk}

w_{jk}	J				
	1	2	3	4	5
30	-0.018264	-0.312366	-0.457487	0.050480	-0.111091

Tabel 4.21 Bobot w_{ok}

w_{ok}	K
	1
30	0.105001

Setelah melakukan perhitungan pada fase propagasi maju, tabel 4.22 menunjukan nilai output(Y) yang dihasilkan:

Tabel 4.22 Hasil output (Y) di badingnkan dengan target (T)

T	Y
0.681201	0.405657
0.637668	0.420486
0.237385	0.440023
0.232297	0.433650
0.167845	0.427134

Dari tabel 4.22, denormalisasi nilai t (target) dan y (output) Menggunakan persamaan 2.21, hasil denormalisasi dapat dilihat pada tabel 4.23:

$$y' = \frac{x' - 0.1}{0.8} (\max - \min) + \min$$

Tabel 4.23 Hasil Denormalisasi t dan y

T	Y
1029.452	541.395
952.3444	567.6608
243.3432	602.2657
234.3311	590.9776
120.1705	579.4361
1029.452	541.395

Hasil denormalisasi akan digunakan untuk menghitung evaluasi ketepatan peramalan menggunakan rumus MAPE (*Mean Absolute Percentage Error*). Berikut persamaan 2.22:

$$MAPE = \frac{1}{n} \sum_i^n \frac{y_k - t}{y_k} \times 100\%$$

$$MAPE = 41,29018\%$$

Dapat dilihat pada proses pengujian di atas, pada iterasi ke-1 dengan jumlah data latih sebesar 30 menghasilkan akurasi sebesar 41,29018%. Proses pelatihan data data latih sebesar 260 akan di terapkan pada program hingga

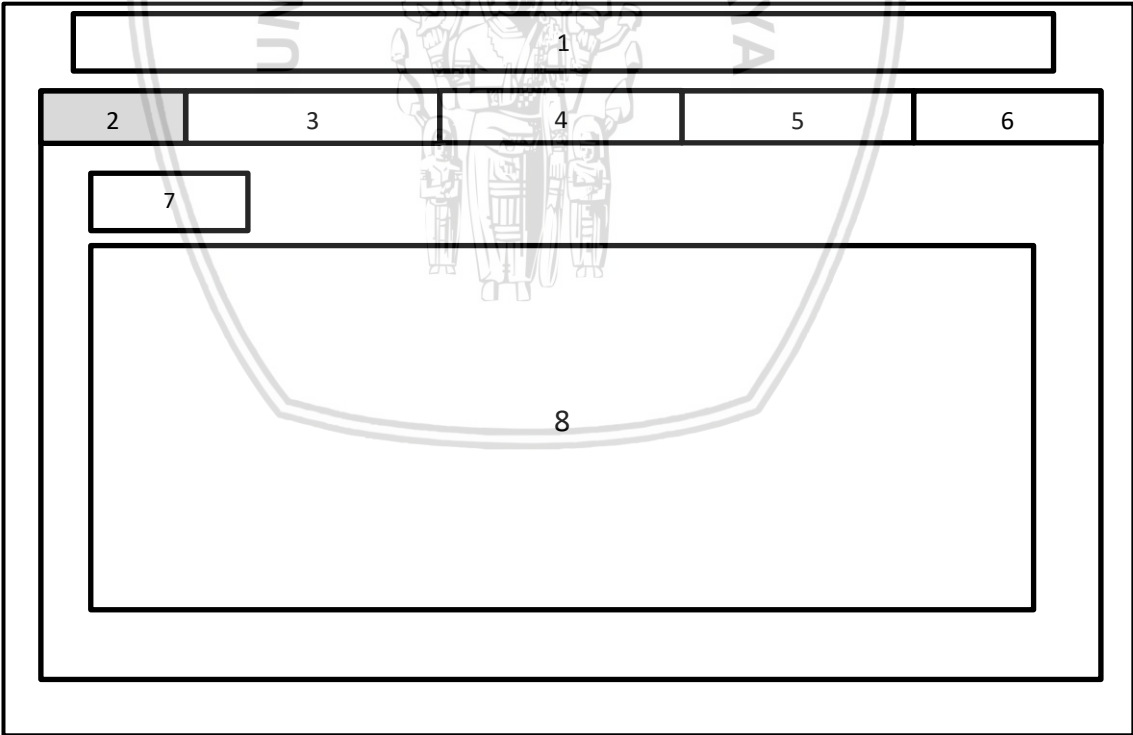
mencapai nilai MAPE yang baik berdasarkan nilai inputan pada proses pelatihan berupa jumlah data latih, jumlah maksimal iterasi, nilai *learning rate*, jumlah *neuron hidden layer* dan batas nilai error.

4.4 Perancangan *User Interface* (Antarmuka)

Interface(Antarmuka) adalah tampilan yang di lihat oleh pengguna yang berfungsi sebagai alat menjembatani komunikasi antar sistem dan pengguna. Perancangan ini dilakukan untuk membantu tahap implementasi nantinya. Pada aplikasi ini terdapat 6 halamam menu yaitu menu login admin dan pengunjung, menu data aktual, menu normalisasi data, menu pelatihan data, menu uji data, dan menu peramalan produksi. Halaman yang dapat di akses admin meliputi semua halaman, sedangkan pengunjung hanya dapat mengakses halaman peramalan produksi saja.

4.4.1 Perancangan Halaman Data Aktual

Halaman ini merupakan halaman untuk menampilkan data keseluruhan yang berjumlah 319 data yang akan digunakan untuk prose spelatihan dan proses pengujian. Ketika pengguna telah berhasil menjalankan perangkat lunak maka halaman ini yang akan tampil terlebih dahulu. Pengguna cukup mengklik button load data untuk menampilkan keseluruhan data aktual.



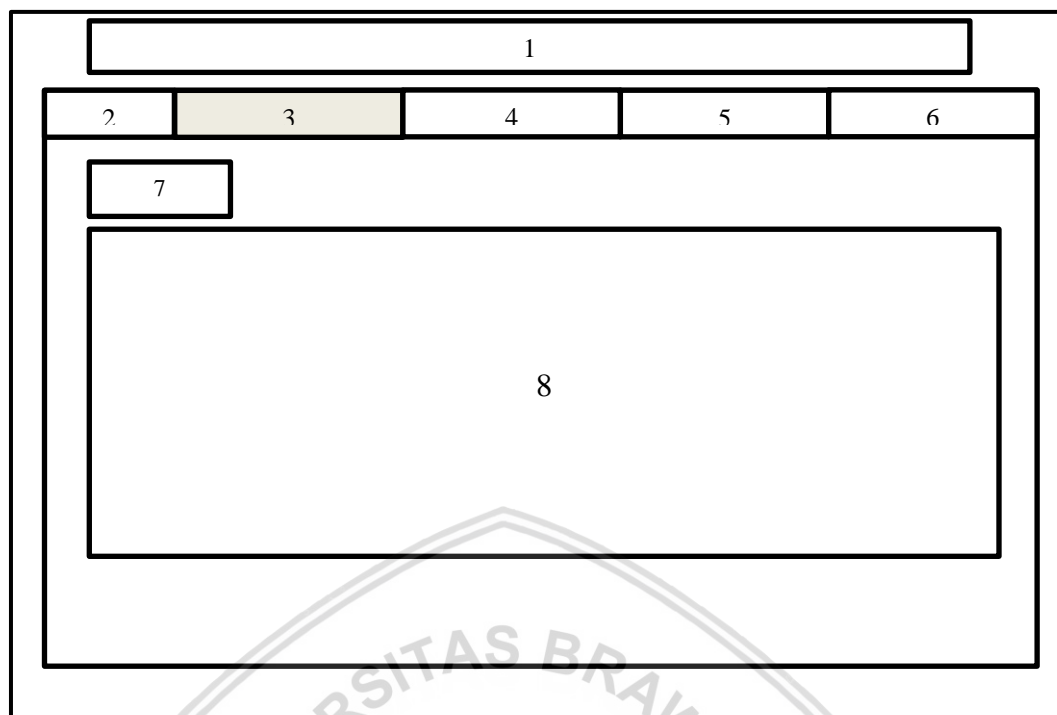
Gambar 4.13 Perancangan Halaman Data Aktual

Tabel 4.24 Halaman Data Aktual

No	Nama	Tipe	Keterangan
1	Peramalan Produksi Kelapa Sawit di PT. Sandabi Indah Lestari	Label	Judul Perangkat Lunak pada penelitian ini
2	Data Aktual	Menu	Pada halaman ini data produksi kelapa sawit dari tahun 2015 hingga 2017 di tampilkan pada tabel
3	Normalisasi	Menu	Pada halaman ini data Produksi di normalisasi dan di tampilkan pada tabel normalisasi data
4	Pelatihan	Menu	Pada halaman ini pelatihan menggunakan algoritme backpropagation dilakukan, ketika proses pelatihan selesai data yang telah di latih di tampilkan pada tabel pelatihan
5	Pengujian	Menu	Pada halaman ini bobot terbaik yang didapatkan pada pelatihan di uji dengan data uji yang telah tersedia
6	Peramalan	Menu	Pada halaman ini peramalan dilakukan dengan menginputkan 6 fitur data yang telah tersedia
7	Load Data	Button	Berfungsi menampilkan data yang ada pada database
8	Tabel Data	Tabel	Tempat tampilnya data setelah button load data di tekan.

4.4.2 Perancangan Halaman Normalisasi Data

Halaman ini berfungsi untuk menampilkan hasil normalisasi data aktual yang telah di tampilkan pada menu Data Aktual. pengguna cukup mengklik button normalisasi data, maka data aktual akan ternormalisasi sesuai dengan menggunakan persamaan 2.17. Gambar 4.14 berikut menampilkan perancangan halaman Normalisasi Data.



Gambar 4.14 Perancangan Halaman Normalisasi Data

Tabel 4.25 Halaman Normalisasi Data

No	Nama	Tipe	Keterangan
1	Peramalan Produksi Kelapa Sawit di PT. Sandabi Indah Lestari	Label	Judul Perangkat Lunak pada penelitian ini
2	Data Aktual	Menu	Pada halaman ini data produksi kelapa sawit dari tahun 2015 hingga 2017 di tampilkan pada tabel
3	Normalisasi	Menu	Pada halaman ini data Produksi di normalisasi dan di tampilkan pada tabel normalisasi data
4	Pelatihan	Menu	Pada halaman ini pelatihan menggunakan algoritme backpropogation dilakukan, ketika proses pelatihan selesai data yang telah di latih di tampilkan pada tabel pelatihan
5	Pengujian	Menu	Pada halaman ini bobot terbaik yang didapatkan pada pelatihan di uji dengan data uji yang telah tersedia
6	Peramalan	Menu	Pada halaman ini peramalan

			dilakukan dengan menginputkan 6 fitur data yang telah tersedia
7	Normalisasi	Button	Berfungsi menampilkan data yang telah di normalisasi
8	Tabel Normalisasi	Tabel	Tempat tampilnya data normalisasi setelah button normalisasi di tekan.

4.4.3 Perancangan Halaman Pelatihan

Halaman ini meruakan halaman pelatihan menggunakan algoritme *backpropagation*, yang terdiri atas fase propagasi maju, propagasi mundur dan perubahan bobot. Untuk melakukan pelatihan data admin harus menginputkan nilai *learning rate*, maksimal iterasi, dan batasan error. Berikut gambar 4.15 rancangan halaman Pelatihan.

Gambar 4.15 Perancangan Halaman Pelatihan

Tabel 4.26 Halaman Pelatihan

No	Nama	Tipe	Keterangan
1	Peramalan Produksi Kelapa Sawit di PT. Sandabi Indah Lestari	Label	Judul Perangkat Lunak pada penelitian ini
2	Data Aktual	Menu	Pada halaman ini data produksi kelapa sawit dari tahun 2015 hingga 2017 di tampilkan pada tabel
3	Normalisasi	Menu	Pada halaman ini data Produksi di normalisasi dan di tampilkan pada tabel normalisasi data
4	Pelatihan	Menu	Pada halaman ini pelatihan menggunakan algoritme backpropagation dilakukan, ketika proses pelatihan selesai data yang telah di latih di tampilkan pada tabel pelatihan
5	Pengujian	Menu	Pada halaman ini bobot terbaik yang didapatkan pada pelatihan di uji dengan data uji yang telah tersedia
6	Peramalan	Menu	Pada halaman ini peramalan dilakukan dengan menginputkan 6 fitur data yang telah tersedia
7	Normalisasi	Button	Berfungsi menampilkan data yang telah di normalisasi
8	Tabel Normalisasi	Tabel	Tempat tampilnya data normalisasi setelah button normalisasi di tekan.
9	Jumlah data	TextField	Tempat inputan nilai jumlah data yang ingin di latih
10	Iterasi	TextField	Tempat inputan batas maksimal iterasi yang dilakukan selama pelatihan data
11	<i>Learning Rate</i>	TextField	Tempat Inputan Nilai laju pembelajaran yang membantu proses pelatihan data
12	Batas Error	TextField	Tempat inputan batas nilai MSE yang ingin di capai
13	MSE dan Iterasi	TextArea	Informasi output berupa nilai MSE pada iterasi ke-n
14	Tabel Pelatihan	Tabel	Tempat di tampilkan data hasil pelatihan

4.4.4 Perancangan Halaman Pengujian

Halaman ini merupakan halaman proses pengujian yang meliputi pengujian beberapa data latih dengan data uji, pengujian nilai *learning rate*, pengujian jumlah iterasi. Pada halaman ini terdapat inputan berupa jumlah data uji, nilai *learning rate*, dan jumlah iterasi.

Gambar 4.16 Perancangan Halaman Pengujian

Tabel 4.27 Halaman Pengujian

No	Nama	Tipe	Keterangan
1	Peramalan Produksi Kelapa Sawit di PT. Sandabi Indah Lestari	Label	Judul Perangkat Lunak pada penelitian ini
2	Data Aktual	Menu	Pada halaman ini data produksi kelapa sawit dari tahun 2015 hingga 2017 di tampilkan pada tabel
3	Normalisasi	Menu	Pada halaman ini data Produksi di normalisasi dan di tampilkan pada tabel normalisasi data
4	Pelatihan	Menu	Pada halaman ini pelatihan menggunakan algoritme backpropagation dilakukan, ketika proses pelatihan selesai data yang telah di latih di tampilkan pada tabel pelatihan

5	Pengujian	Menu	Pada halaman ini bobot terbaik yang didapatkan pada pelatihan di uji dengan data uji yang telah tersedia
6	Peramalan	Menu	Pada halaman ini peramalan dilakukan dengan menginputkan 6 fitur data yang telah tersedia
7	Jumlah Data Uji	TextField	Tempat inputan jumlah data uji yang akan di uji
8	Uji Data	Button	Berfungsi menampilkan hasil pengujian pada tabel pengujian
9	MAPE dan akurasi	TextArea	Berisi informasi nilai MAPE dan akurasi
10	Tabel Pengujian	Tabel	Tempat data hasil pengujian di tampilkan

4.4.5 Perancangan Halaman Peramalan Produksi

Halaman Peramalan merupakan halaman hasil peramalan jumlah produksi setelah melalui proses algoritme *backpropagation*. Pada halaman ini tersedia inputan umur kelapa sawit, jumlah pokok kelapa sawit, luas lahan kelapa sawit, produksi 3 bulan sebelumnya, produksi 2 bulan sebelumnya, dan produksi 1 bulan sebelumnya.

The diagram illustrates the layout of the Production Forecasting Page. It features a header bar (1) and a navigation menu (2, 3, 4, 5, 6). The main content area includes input fields for 'Umur' (7), 'Jumlah Pokok' (8), 'Luas Lahan' (9), 'Produksi bulan pertama' (10), 'Produksi bulan kedua' (11), and 'Produksi bulan ketiga' (12). A large table (15) displays 'Data Produksi Tahun 2015 - 2017'. Other elements include a '13' label, a '14' label, and a '15' label.

Gambar 4.17 Perancangan Halaman Peramalan

Tabel 4.28 Halaman Peramalan Produksi

No	Nama	Tipe	Keterangan
1	Peramalan Produksi Kelapa Sawit di PT. Sandabi Indah Lestari	Label	Judul Perangkat Lunak pada penelitian ini
2	Data Aktual	Menu	Pada halaman ini data produksi kelapa sawit dari tahun 2015 hingga 2017 di tampilkan pada tabel
3	Normalisasi	Menu	Pada halaman ini data Produksi di normalisasi dan di tampilkan pada tabel normalisasi data
4	Pelatihan	Menu	Pada halaman ini pelatihan menggunakan algoritme backpropagation dilakukan, ketika proses pelatihan selesai data yang telah di latih di tampilkan pada tabel pelatihan
5	Pengujian	Menu	Pada halaman ini bobot terbaik yang didapatkan pada pelatihan di uji dengan data uji yang telah tersedia
6	Peramalan	Menu	Pada halaman ini peramalan dilakukan dengan menginputkan 6 fitur data yang telah tersedia
7	Umur	TextField	Tempat inputan parameter umur
8	Jumlah Pokok	TextField	Tempat inputan parameter jumlah pokok
9	Luas Lahan	TextField	Tempat inputan parameter luas lahan
10	Produksi bulan pertama	TextField	Tempat inputan parameter produksi bulan pertama
11	Produksi bulan kedua	TextField	Tempat inputan parameter produksi bulan kedua
12	Produksi bulan ketiga	TextField	Tempat inputan parameter produksi bulan ketiga
13	Ramal	Button	Berfungsi memproses peramalan dan menampilkan hasil ramalan
14	Tabel Hasil Ramal	Tabel	Tempat hasil ramalan di tampilkan
15	Tabel Data Produksi	Tabel	Tempat data produksi kelapa swit PT. Sandabi Indah Lestari di tampilkan dari bulan januari tahun 2015 hingga bulan agustus tahun 2017

4.5 Perancangan Pengujian

Pada sub bab ini akan dilakukan pengujian terhadap algoritme *backpropagation* yang digunakan pada penelitian peramalan produksi kelapa sawit PT. Sandabi Indah Lestari. Metode *backpropagation* akan diuji kehandalannya dalam menemukan peramalan yang mendekati target. Beberapa pengujian akan dilakukan yaitu pengujian nilai *learning rate*, pengujian jumlah iterasi, pengujian jumlah data latih, pengujian jumlah *neuron Hidden Layer*, dan pengujian *K-Fold Cross Validation*. berikut perancangan pengujian yang akan dilakukan :

4.5.1 Perancangan Pengujian Jumlah Iterasi Maksimal

Pada pengujian jumlah Iterasi maksimal percobaan dilakukan sebanyak 5 kali, data latih yang digunakan sebesar 260 data, data uji yang digunakan berjumlah 59 data, nilai learning rate yang digunakan sebesar 0.4, *neuron hidden layer* berjumlah 5 dan batas error sebesar 0.001. Adapun jumlah iterasi maksimal yang digunakan terdiri atas 200, 300, 400, 500, 600, 700, 800, 900, 1000, 2000. Pengujian iterasi maksimal bertujuan mendapatkan nilai iterasi maksimal terbaik yang menghasilkan nilai MAPE terbaik. Berikut tabel 4.29 rancangan pengujian Jumlah Iterasi Maksimal.

Tabel 4.29 Perancangan Pengujian Jumlah Iterasi Maksimal

Jumlah Iterasi Maksimal	Nilai MAPE (%) pada percobaan ke-n					Rata- rata Nilai MAPE (%)
	1	2	3	4	5	
200						
300						
400						
500						
600						
700						
800						
900						
1000						
2000						
Rata- Rata MAPE (%) Keseluruhan						

4.5.2 Perancangan Pengujian Jumlah Data Latih

Pada perancangan pengujian jumlah data latih percobaan dilakukan sebanyak 5 kali, data uji berjumlah 40 data, nilai learning rate 0.4, iterasi maksimal sejumlah 900, *neuron hidden layer* berjumlah 5 dan batas error 0.001. umlah data latih yang di uji di antaranya 52, 104, 156, 208, dan 260. Pengujian ini bertujuan menentukan jumlah data latih yang menghasilkan nilai MAPE terbaik. Berikut tabel 4.36 rancangan pengujian jumlah data latih.

Tabel 4.30 Perancangan Pengujian Jumlah Data Latih

Jumlah data latih	Nilai MAPE (%) pada percobaan ke-n					Rata- rata Nilai MAPE (%)
	1	2	3	4	5	
52						
104						
156						
208						
260						
Rata- Rata MAPE (%) Keseluruhan						

4.5.3 Perancangan Pengujian Nilai *Learning Rate*

Pada pengujian nilai *learning rate* percobaan dilakukan sebanyak 5 kali ,jumlah data latih yang digunakan sebesar 260 data ,data uji yang digunakan berjumlah 59 data, iterasi maksimal sebesar 1000, *neuron hidden layer* berjumlah 5 dan batas error sebesar 0.001. Nilai *learning rate* yang di uji terdiri dari 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7. Pengujian ini bertujuan untuk mendapatkan nilai *learning rate* yang menghasilkan Nilai MAPE terbaik. Berikut tabel 4.31 rancangan pengujian nilai *learning rate*.

Tabel 4.31 Perancangan Pengujian Nilai *Learning Rate*

Nilai Learning Rate	Nilai MAPE (%) pada percobaan ke-n					Rata- rata Nilai MAPE (%)
	1	2	3	4	5	
0.1						
0.2						
0.3						
0.4						
0.5						
0.6						

0.7						
Rata- Rata MAPE (%) Keseluruhan						

4.5.4 Perancangan Pengujian Jumlah *Neuron Hidden Layer*

Pada pengujian jumlah *Neuron Hidden Layer* percobaan dilakukan sebanyak 5 kali ,dengan data latih yang sebesar 260 data, data uji yang digunakan berjumlah 59 data, nilai learning rate yang digunakan sebesar 0.4, iterasi maksimal sebesar 900 dan batas error 0.001. jumlah *neuron hidden layer* yang diuji mulai dari 1 hingga 5. Pengujian ini bertujuan menemukan struktur *neuron hidden layer* terbaik yang menghasilkan nilai MAPE seminimal mungkin. Berikut tabel 4.32 rancangan pengujian jumlah *neuron hidden layer*.

Tabel 4.32 Perancangan Pengujian Jumlah *Neuron Hidden Layer*

Jumlah neuron	Nilai MAPE (%) pada percobaan ke-n					Rata- rata Nilai MAPE (%)
	1	2	3	4	5	
1						
2						
3						
4						
5						
Rata- Rata MAPE (%) Keseluruhan						

4.5.5 Perancangan Pengujian *K-Fold Cross Validation*

Pada pengujian *K-Fold Cross Validation* data latih dan data uji di kelompokkan berdasarkan nilai k, jumlah data keseluruhan yang digunakan pada penelitian ini sebesar 319 data. Nilai K-Fold yang digunakan pada penelitian ini yaitu 5 dan 10. Pada K-5, data latih di kelompokkan menjadi 5 kelompok dengan jumlah 66 data setiap kelompok, kemudian pada k-7 data dikelompokkan menjadi 7 kelompok dengan jumlah 44 data setiap kelompok. Berikut gambar perancangan pengujian *K-Fold Cross Validation*.

Fold ke 1 dengan nilai K = 5						
66	66	66	66	55		
Fold ke 1 dengan nilai K = 7						
44	44	44	44	44	44	55

Gambar 4.18 Perancangan K-Fold

Pada gambar di atas menunjukan Fold ke 1 dengan K-5 dan K-7. Nilai yg di arsisr pada K-5 merupakan data uji yaitu data dengan urutan nomor 1 hingga 66, sedangkan data dengan urutan nomor 67 hingga 319 menjadi data latih.

Kemudian Nilai yang di arsir pada K-7 merupakan data uji dengan urutan nomor 1 hingga 44, kemudian data dengan urutan nomor 45 hingga 319 merupakan data latih. Pada fold 2 dengan K-5 data uji yang digunakan mulai dari urutan nomor 67 hingga 132 dan sisanya menjadi data latih. Proses fold selanjutnya mengikuti aturan yang telah dijelaskan. Percobaan dilakukan sebanyak 5 kali dengan nilai learning rate 0.4, iterasi maksimal sejumlah 900, *neuron hidden layer* berjumlah 5 dan batas error 0.001. berikut tabel 4.33 dan 4.34 perancangan pengujian *K-Fold Cross Validation*.

Tabel 4.33 Perancangan Pengujian *K-Fold Cross Validation* dengan nilai k=5

Fold	Nilai MAPE (%) pada percobaan ke-n					Rata- rata Nilai MAPE (%)
	1	2	3	4	5	
1						
2						
3						
4						
5						
Rata- Rata MAPE (%) Keseluruhan						

Tabel 4.34 Perancangan Pengujian *K-Fold Cross Validation* dengan nilai k=7

Fold	Nilai MAPE (%) pada percobaan ke-n					Rata- rata Nilai MAPE (%)
	1	2	3	4	5	
1						
2						
3						
4						
5						
6						
7						
Rata- Rata MAPE (%) Keseluruhan						

BAB 5 IMPLEMENTASI

Pada bab ini menjelaskan tentang implemetasi sistem yang telah di rancang pada bab sebelumnya.

5.1 Implementasi Sistem

Perangkat lunak di implemetasikan menggunakan bahasa pemrograman java dengan lingkungan Sistem Operasi Windows 10 dan aplikasi Netbeans sebagai tools pendukung mengimplemetasikan bahasa pemrograman java.

5.1.1 Implementasi Normalisasi data

```

1 public double[][] norm() {
2     for (int j = 0; j < data.length; j++) {
3         normData[j][0] = (((((double) data[j][0]) - min1()) /
4 (max1() - min1())) * 0.8) + 0.1;
5         normData[j][1] = (((((double) data[j][1]) - min2()) /
6 (max2() - min2())) * 0.8) + 0.1;
7         normData[j][2] = (((((double) data[j][2]) - min3()) /
8 (max3() - min3())) * 0.8) + 0.1;
9         normData[j][3] = (((((double) data[j][3]) - min4()) /
10 (max4() - min4())) * 0.8) + 0.1;
11        normData[j][4] = (((((double) data[j][4]) - min5()) /
12 (max5() - min5())) * 0.8) + 0.1;
13        normData[j][5] = (((((double) data[j][5]) - min6()) /
14 (max6() - min6())) * 0.8) + 0.1;
15        normData[j][6] = (((((double) data[j][6]) - min7()) /
16 (max7() - min7())) * 0.8) + 0.1;
17    }
18    return normData;
19 }

```

Kode Program 5.1 Normalisasi Data

Penjelasan Kode Program 5.1:

1. Baris 1 nama method dengan tipe double Array 2 dimensi
2. Baris 2 perulangan untuk melakukan normalisasi pada semua data latih sebanyak panjang data yaitu 260 dats
3. Baris 3-16 proses normalisasi data pada parameter umur, jumlah pokok, luas lahan, hasil panen bulan pertama, hasil panen bulan kedua, hasil panen bulan ketiga serta target yang merupakan hasil panen bulan keempat.

5.1.2 Implementasi Inisialisasi Bobot Awal

```

1 public double[][] bobotVij() {
2     for (int i = 0; i < jumHidden; i++) {
3         for (int j = 0; j < jumInput; j++) {
4             V[i][j] = min
5                 + (max - min) *
6             random.nextDouble();
7         }
8     }
9 }

```



```

8         System.out.println("");
9     }
10    return V;
11 }
12 //algoritme nguyen widrow
13 public double[] vektor() {
14     for (int i = 0; i < jumHidden; i++) {
15         vektor[i] = Math.pow(V[i][0], 2)
16             + Math.pow(V[i][1], 2)
17             + Math.pow(V[i][2], 2)
18             + Math.pow(V[i][3], 2)
19             + Math.pow(V[i][4], 2)
20             + Math.pow(V[i][5], 2);
21     }
22     return vektor;
23 }
24
25 public double[] vektorVij() {
26     for (int i = 0; i < jumHidden; i++) {
27         vektorV[i] = Math.sqrt(vektor[i]);
28     }
29     return vektorV;
30 }
31 public double nilaiFSkala() {
32     faktorSkala = 0.7 *
33     Math.pow(Math.abs(jumHidden), 1 / jumInput);
34     return faktorSkala;
35 }
36
37 public double[][] bobotVijbaru() {
38     for (int j = 0; j < jumHidden; j++) {
39         for (int i = 0; i < jumInput; i++) {
40
41             Vbaru[j][i] = (nilaiFSkala()
42                 * V[j][i]) / vektorV[j];
43         }
44         System.out.println("");
45     }
46     return Vbaru;
47 }
48
49 public double[][] bobotVoj() {
50     for (int o = 0; o < jumData; o++) {
51         for (int j = 0; j < jumHidden; j++) {
52             biasV[o][j] = -nilaiFSkala() + (nilaiFSkala()
53                 - (-nilaiFSkala())) *
54             random.nextDouble();
55         }
56     }
57     return biasV;
58 }
59
60 }

```

```

61
62     public double[][] bobotWjk() {
63         for (int i = 0; i < jumData; i++) {
64             for (int j = 0; j < jumHidden; j++) {
65                 W[i][j] = min + (max - min) * random.nextDouble();
66             }
67         }
68         return W;
69     }
70
71     public double[][] bobotWok() {
72         for (int i = 0; i < jumData; i++) {
73             biasW[i][0] = min + (max - min) *
74 random.nextDouble();
75         }
76         return biasW;
77     }

```

Kode Program 5.2 Inisialisasi Bobot Awal

Penjelasan Kode Program 5.2:

1. Baris 1-17 inisialisasi bobot awal Vij.
2. Baris 19-36 proses perhitungan vector dari nilai awal Vij
3. Baris 37-42 proses perhitungan nilai Faktor Skala pada algoritme *nguyen Widrow*
4. Baris 44-56 proses inisialisasi bobot Vij yang baru dari algoritme *nguyen widrow*
5. Baris 57-60 Proses inisialisasi bobot Voj
6. Baris 62-69 Proses inisialisasi bobot Wjk
7. Baris 71-77 Proses inisialisasi bobot Wok

5.1.3 Implementasi Fase Propagasi Maju

```

1     public double[][] siyalZin() {
2         double jumXV = 0;
3         for (int o = 0; o < jumData; o++) {
4             for (int j = 0; j < jumHidden; j++) {
5                 for (int i = 0; i < jumInput; i++) {
6                     double tmp = n.normData[o][i] * Vbaru[j][i];
7                     jumXV = jumXV + tmp;
8                 }
9                 Zin[o][j] = biasV[o][j] + jumXV;
10            }
11        }
12        return Zin;
13    }
14
15    public double[][] sinyalYin() {
16        double jumWZ = 0;
17        for (int o = 0; o < jumData; o++) {
18            for (int j = 0; j < jumHidden; j++) {
19                double temp = W[o][j] * Z[o][j];
20                jumWZ = jumWZ + temp;

```

```

21         }
22         Yin[o][0] = biasW[o][0] + jumWZ;
23     }
24     return Yin;
25 }
26 public double[][] aktivasiZin() {
27     for (int o = 0; o < jumData; o++) {
28         for (int j = 0; j < jumHidden; j++) {
29             Z[o][j] = 1 / (1 + (double) Math.exp(-
30 Zin[o][j]));
31         }
32     }
33     return Z;
34 }
35
36 public double[][] aktivasiYin() {
37     for (int o = 0; o < jumData; o++) {
38         Y[o][0] = 1 / (1 + (double) Math.exp(-
39 Yin[o][0]));
40     }
41     return Y;
42 }

```

Kode Program 5.3 Fase Propagasi Maju

Penjelasan Kode Program 5.3:

1. Baris 1-13 Proses perhitungan sinyal Zin
2. Baris 15-25 proses perhitungan sinyal Yin
3. Baris 26-34 Proses Aktivasi sinyal Zin yang menghasilkan nilai Z
4. Baris 36-42 Proses Aktivasi sinyal Yin yang menghasilkan nilai Y

5.1.4 Implementasi Fase Propagasi Mundur

```

1 public double[][] errorY() {
2     for (int o = 0; o < jumData; o++) {
3         errorY[o][0] = (n.normData[o][6] - Y[o][0])
4 * Y[o][0] * (1 - Y[o][0]);
5     }
6     return errorY;
7 }
8 public double[][] deltaVij() {
9     for (int i = 0; i < jumData; i++) {
10         for (int j = 0; j < jumHidden; j++) {
11             deltaV[i][j] = learningRate *
12 errorZ[i][j] * n.normData[i][j];
13         }
14     }
15     return deltaV;
16 }
17
18 public double[][] deltaVoj() {
19     for (int i = 0; i < jumData; i++) {
20         for (int j = 0; j < jumHidden; j++) {
21             deltaBiasV[i][j] = learningRate *

```

```

22 errorZ[i][j];
23     }
24     }
25     return deltaBiasV;
26 }
27
28 public double[][] deltaWok() {
29     for (int o = 0; o < jumData; o++) {
30         deltaBiasW[o][0] = learningRate *
31 errorY[o][0];
32     }
33     return deltaBiasW;
34 }
35
36 public double[][] deltaWjk() {
37     for (int o = 0; o < jumData; o++) {
38         for (int j = 0; j < jumHidden; j++) {
39             deltaW[o][j] = learningRate *
40 errorY[o][0] * Z[o][j];
41         }
42     }
43     return deltaW;
44 }
45
46 public double[][] errorZin() {
47     int n = 0;
48     for (int i = 0; i < jumData; i++) {
49         for (int j = 0; j < jumHidden; j++) {
50             errorZin[i][j] = errorY[i][n] * W[i][j];
51         }
52     }
53     return errorZin;
54 }
55
56 public double[][] errorZ() {
57     for (int i = 0; i < jumData; i++) {
58         for (int j = 0; j < jumHidden; j++) {
59             errorZ[i][j] = errorZin[i][j] * Z[i][j]
60 * (1 - Z[i][j]);
61         }
62     }
63     return errorZ;
64 }
65
66 public double[][] errorT() {
67     for (int i = 0; i < jumData; i++) {
68         errorT[i][0] = n.normData[i][6] - Y[i][0];
69     }
70     return errorT;
71 }

```

Kode Program 5.4 Fase Propagasi Mundur

Penjelasan Kode Program 5.4:

1. baris 1-7 proses perhitungan nilai koreksi error Y.
2. baris 8-16 proses perhitungan ΔV_{ij}
3. baris 18-24 proses perhitungan ΔV_{oj}
4. baris 28-34 proses perhitungan ΔW_{ok}
5. baris 36-44 proses perhitungan ΔW_{jk}
6. baris 46-54 Proses perhitungan koreksi error Zin
7. baris 56-64 proses perhitungan koreksi error Z
8. baris 66-71 proses perhitungan Error

5.1.5 Ubah Bobot

```

1  public double[][] ubahVij() {
2      bobotVijbaru();
3      for (int o = 0; o < jumData; o++) {
4          for (int j = 0; j < jumHidden; j++) {
5              for (int i = 0; i < jumInput; i++) {
6                  Vbaru[j][i] = Vbaru[j][i] + deltaV[o][j];
7              }
8          }
9      }
10     return Vbaru;
11 }
12
13 public double[][] ubahVoj() {
14     for (int i = 0; i < jumData; i++) {
15         for (int j = 0; j < jumHidden; j++) {
16             biasV[i][j] = biasV[i][j] + deltaBiasV[i][j];
17         }
18     }
19     return biasV;
20 }
21
22 public double[][] ubahWok() {
23     for (int i = 0; i < jumData; i++) {
24         for (int j = 0; j < jumOutput; j++) {
25             biasW[i][j] = biasW[i][j] +
26 deltaBiasW[i][j];
27         }
28     }
29     return biasW;
30 }
31
32 public double[][] ubahWjk() {
33     for (int i = 0; i < jumData; i++) {
34         for (int j = 0; j < jumHidden; j++) {
35             W[i][j] = W[i][j] + deltaW[i][j];
36         }
37     }
38     return W;
39 }

```

Kode Program 5.5 Ubah Bobot

Penjelasan Kode Program 5.5:

1. Baris 1-11 proses perhitungan ubah bobot V_{ij}
2. Baris 13-20 proses perhitungan ubah bobot V_{oj}
3. Baris 22-30 proses perhitungan ubah bobot W_{ok}
4. Baris 32-40 proses perhitungan ubah bobot W_{jk}

5.1.6 implementasi Pengujian

```

1 //Fase Propagasi Maju
2 public double[][] sinyalZin() {
3     double jumXV = 0;
4     for (int i = 0; i < jumDataUji; i++) {
5         for (int k = 0; k < jumHidden; k++) {
6             for (int j = 0; j < jumInput; j++) {
7                 double tmp = normData[i][j] * a.Vbaru[k][j];
8                 jumXV = jumXV + tmp;
9             }
10            Zin[i][k] = a.biasV[i][k] + jumXV;
11        }
12    }
13    return Zin;
14 }
15 public double[][] sinyalYin() {
16     double jumWZ = 0;
17     for (int i = 0; i < jumDataUji; i++) {
18         for (int j = 0; j < jumHidden; j++) {
19             double temp = a.W[i][j] * Z[i][j];
20             jumWZ = jumWZ + temp;
21         }
22         Yin[i][0] = a.biasW[i][0] + jumWZ;
23     }
24    return Yin;
25 }
26
27 public double[][] aktivasiZin() {
28     for (int i = 0; i < jumDataUji; i++) {
29         for (int j = 0; j < jumHidden; j++) {
30             Z[i][j] = 1 / (1 + (double) Math.exp(-Zin[i][j]));
31         }
32     }
33    return Z;
34 }
35
36 public double[][] aktivasiYin() {
37     for (int i = 0; i < jumDataUji; i++) {
38         Yin[i][0] = 1 / (1 + (double) Math.exp(-Yin[i][0]));
39     }
40    return Y;
41 }
42
43
44 public double[][] DenormalisasiY() {

```

```

45         for (int i = 0; i < jumDataUji; i++) {
46             denormDataY[i][0] = ((Y[i][0] - 0.1) / 0.8) *
47             ((max7 - min7) + min7);
48         }
49         return denormDataY;
50     }
51
52     public double MAPE(int dt) {
53         setJumDataUji(dt);
54         double jumMAPE = 0;
55         for (int i = 0; i < jumDataUji; i++) {
56             double temp = Math.abs((((denormDataY[i][0] -
57             denormData[i][0]) / denormDataY[i][0]) * 100));
58             jumMAPE = jumMAPE + temp;
59         }
60         MAPE = jumMAPE / jumDataUji;
61         return MAPE;
62     }

```

Kode Program 5.6 Pengujian

Penjelasan Kode Program 5.6:

1. baris 2-13 proses perhitungan sinyal Zin
2. baris 15-25 perhitungan sinyal Yin
3. baris 27-34 perhitungan aktivasi sinyal Zin menjadi nilai Z
4. baris 36-42 perhitungan aktivasi sinyal Yin menjadi nilai Y
5. baris 44-50 perhitungan denormalisasi nilai Y
6. baris 52-62 perhitungan MAPE

5.1.7 Implementasi Peramalan

```

1  public void setParameter(int umur, int jumPokok, int
2  luasLahan, int bulan3, int bulan2, int bulan1) {
3
4      this.umur = umur;
5      this.jumPokok = jumPokok;
6      this.luasLahan = luasLahan;
7      this.t3 = bulan3;
8      this.t2 = bulan2;
9      this.t1 = bulan1;
10
11  }
12
13  public double[][] DenormalisasiY(int umur, int jumPokok,
14  int luasLahan, int bulan3, int bulan2, int bulan1) {
15      for (int j = 0; j < norm.data.length; j++) {
16          if (umur == norm.data[j][0] && jumPokok ==
17          norm.data[j][1] && luasLahan == norm.data[j][2] &&
18          bulan3 == norm.data[j][3] && bulan2 == norm.data[j][4]
19          && bulan1 == norm.data[j][5]) {
20
21          denormDataY[0][0] = ((a.Y[j][0] - 0.1) / 0.8) *

```



```

22 ((norm.max7() - norm.min7()) + norm.min7());
23     }
24 }
25 }
26 return denormDataY;
27 }
28 }
29
30
31 public void peramalan(int jumdataLatih, int iterasi,
32 double r, int hidden,
33     double e, int umur, int jumPokok, int
34 luasLahan, int bulan3, int bulan2, int bulan1) {
35     norm.normalisasi();
36     a.latih(jumdataLatih, iterasi, r, hidden,e);
37     DenormalisasiY(umur,jumPokok, luasLahan, bulan3,
38 bulan2,bulan1);
39 }
40
41
42

```

Kode Program 5.7 Peramalan

Penjelasan Kode Program 5.7:

1. Baris 1-12 mengeset parameter method sebagai variable pada class peramalan.
2. Baris 13-28 proses denormalisasi Y sebagai nilai ramalan.
3. Baris 31-42 proses peramalan mulai dari menjalankan method normalisasi() , method latih() dan menjalankan method denormalisasiY().

5.2 Implementasi *User Interface*

Hasil implementasi menggunakan *Graphic User Interface* (GUI) dengan bahasa java yang tersedia pada Netbeans. *User Interface* yang di implementasikan antara lain data aktual, normalisasi, pelatihan, pengujian dan peramalan.

5.2.1 Implementasi *User Interface* Data Aktual

User Interface Data Aktual menampilkan data keseluruhan yang digunakan untuk data latih serta data uji. Hasil implementasi *User Interface* Data Aktual dapat di lihat pada gambar 5.1.

Umur (Tahun)	Jumlah Pokok	Luas Lahan (Hektar)	Hasil Panen Bulan Pertama	Hasil Panen Bulan Kedua (Ton)	Hasil Panen Bulan Ketiga	Hasil Panen Bulan ke-4 sebagai T
15	2562	18	33	21	23	29
16	2562	18	33	33	39	37
17	2562	18	29	33	37	39
14	33635	288	413	355	402	417
15	33635	288	557	489	533	504
16	33635	288	487	440	525	540
13	21799	185	235	191	214	254
14	21799	185	307	255	340	279
15	21799	185	288	248	300	302
12	87583	716	1913	755	957	993
13	87583	716	1997	1149	1395	1257
14	87583	716	1199	979	1394	1222
11	85510	706	850	723	869	935
12	85510	706	1119	930	1273	1196
13	85510	706	1963	953	1118	1127
10	17970	148	214	170	214	191
11	17970	148	253	245	301	250
12	17970	148	344	236	277	279
9	10583	82	138	109	129	146
10	10583	82	124	102	174	136
11	10583	82	184	117	143	174
8	436	8	4	2	3	5
0	0	0	0	0	0	0

Gambar 5.1 User Interface Data Aktual

5.2.2 Implementasi User Interface Normalisasi

User Interface Normalisasi menampilkan data latih yang telah di normalisasi menggunakan persamaan 2.17. Hasil implementasi User Interface Normalisasi dapat dilihat pada gambar 5.2.

Umur (Tahun)	jumlah pokok	luas lahan (Hektar)	Hasil Panen bulan pertama	Hasil Panen produksi bulan	Hasil Panen produksi bulan	target
0.8058823529411765	0.122675275007707	0.12011173184357543	0.12033033036584746	0.11462140602167102	0.11357933578135794	0.11845564280031823
0.8529411764705882	0.123675275007707	0.12011173184357543	0.12237204135543222	0.12436601653611637	0.1230256305200259	0.123646130466037152
0.9	0.123675275007707	0.12011173184357543	0.119561015934915254	0.12297650130548304	0.12194203848018451	0.12462100238663485
0.7588235294117647	0.4099688239573618	0.41284916201117317	0.38	0.347171453437772	0.33734316734317343	0.36536319474940337
0.8058823529411765	0.4099688239573618	0.41284916201117317	0.4770271185440078	0.4205448215836661	0.4148853459534598	0.42076323215035804
0.8529411764705882	0.4099688239573618	0.41284916201117317	0.4301904915254237	0.4063533507397374	0.4026520936309953	0.44367541766100786
0.7117647058823529	0.29911626989544	0.3067039106145252	0.2593203338863051	0.2299039164490062	0.2213459834886347	0.26165473348944234
0.7588235294117647	0.29911626989544	0.3067039106145252	0.308135593220339	0.2775456919060052	0.26073800739007384	0.2719556324562339
0.8058823529411765	0.29911626989544	0.3067039106145252	0.2952542372881356	0.27263188899876155	0.2771217712177122	0.29220365950616214
0.6547058823529413	0.9	0.9	0.12671735510159349	0.628674489954839	0.6550184501845018	0.7319806069212411
0.7117647058823529	0.9	0.9	0.843728813559322	0.9	0.9	0.9
0.7588235294117647	0.9	0.9	0.9	0.7816352053855695	0.899892989929899	0.8777247414478818
0.8176470588235294	0.8810848195150086	0.8862122905027933	0.7440977999101695	0.6033942258746337	0.6139327306273062	0.6990676213206046
0.8647058823529413	0.8810848195150086	0.8862122905027933	0.852542372881355	0.8111453437711074	0.8515867158671587	0.8605400706648369
0.7117647058823529	0.8810848195150086	0.8862122905027933	0.8206779991016949	0.783532013971372	0.7988929899298993	0.8172532323716836
0.5705882352941177	0.26414144297409314	0.26646044692737427	0.2459547457627119	0.2225413402959095	0.2263468834688347	0.2215592689664757
0.6176470588235294	0.26414144297409314	0.26646044692737427	0.2198474576271199	0.21056311575262856	0.2177121771217712	0.2591088869578157
0.6547058823529413	0.26414144297409314	0.26646044692737427	0.26542372881355936	0.2643187872146695	0.2635424354243543	0.2775656324582339
0.525294117647059	0.1999977914354364	0.20279329608938548	0.19421728813559324	0.17589206006062578	0.17439154391543912	0.19291964966022278
0.5705882352941177	0.1999977914354364	0.20279329608938548	0.1840977999101695	0.18484342900617544	0.2027305273002731	0.18991885441527448
0.6176470588235294	0.1999977914354364	0.20279329608938548	0.1540977999101695	0.18145214099216712	0.18442804428044282	0.21073085680190602
0.4764705882352941	0.10388250802396299	0.10556859217877095	0.10271199440677957	0.10138251523063535	0.10177121771217713	0.10316217979015632
0.1	0.1	0.1	0.1	0.1	0.1	0.1

Gambar 5.2 User Interface Normalisasi

5.2.3 Implementasi User Interface Pelatihan

User Interface Pelatihan menampilkan hasil pelatihan data dengan menginputkan jumlah data latih, maksimal iterasi, nilai learning rate, jumlah

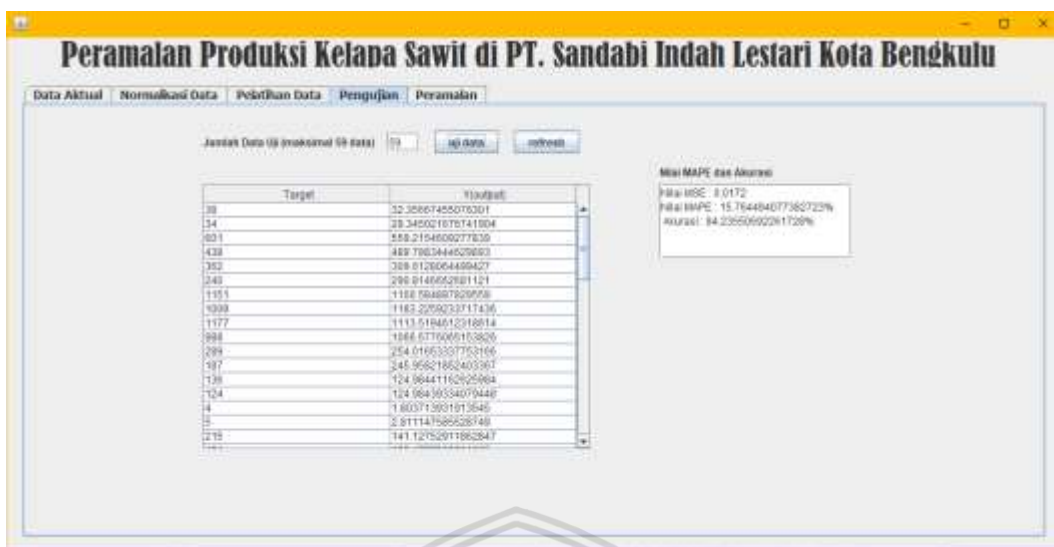
neuron hidden layer, dan batas error. Kemudian ketika mengklik button latih data maka pada tabel akan tampil data hasil pelatihan sesuai dengan inputan yang di masukkan dan pada text area akan menampilkan informasi nilai MSE serta Iterasi terakhir. Hasil implementasi *User Interface* Pelatihan dapat dilihat pada gambar 5.3.



Gambar 5.3 User Interface Pelatihan

5.2.4 Implementasi User Interface Pengujian

User Interface Pengujian menampilkan hasil pengujian dengan menginputkan jumlah data uji, ketika mengklik button uji data maka tabel akan menampilkan hasil pengujian dan text Area akan menampilkan informasi nilai MSE, MAPE, dan Akurasi sistem. Hasil Implementasi *User Interface* Pengujian dapat dilihat pada gambar 5.4.



Gambar 5.4 User Interface Pengujian

5.2.5 Implementasi User Interface Peramalan

User Interface Peramalan menampilkan hasil peramalan dengan menginputkan 6 parameter yaitu umur, jumlah pokok, luas lahan, hasil panen bulan ke pertama, hasil panen bulan kedua, dan hasil panen bulan ketiga, ketika button ramal diklik maka hasil peramalan akan muncul pada tabel. Hasil Implementasi User Interface Peramalan dapat di lihat pada gambar 5.5.



Gambar 5.5 User Interface peramalan

BAB 6 PENGUJIAN DAN ANALISIS HASIL

6.1 Pengujian

Pada bab ini akan menjelaskan hasil pengujian yang telah dilakukan terhadap implementasi algoritme *backpropagation* untuk peramalan produksi kelapa sawit di PT. Sandabi Indah Lestari. Pengujian yang dilakukan sesuai pada bab perancangan sebelumnya yaitu:

1. Pengujian Iterasi Maksimal
2. Pengujian jumlah data latih
3. Pengujian Nilai Learning Rate
4. Pengujian jumlah *neuron hidden layer*
5. Pengujian *K-fold Cross Validation*

Pengujian diatas berdasarkan nilai MAPE yang dihasilkan. Nilai MAPE terbaik pada setiap pengujian akan digunakan untuk mendapatkan hasil peramalan terbaik pada analisis akhir. Hasil kelima pengujian akan dijadikan referensi dalam menjelaskan sub bab analisis hasil pengujian.

7.1.1 Hasil Pengujian Iterasi Maksimal

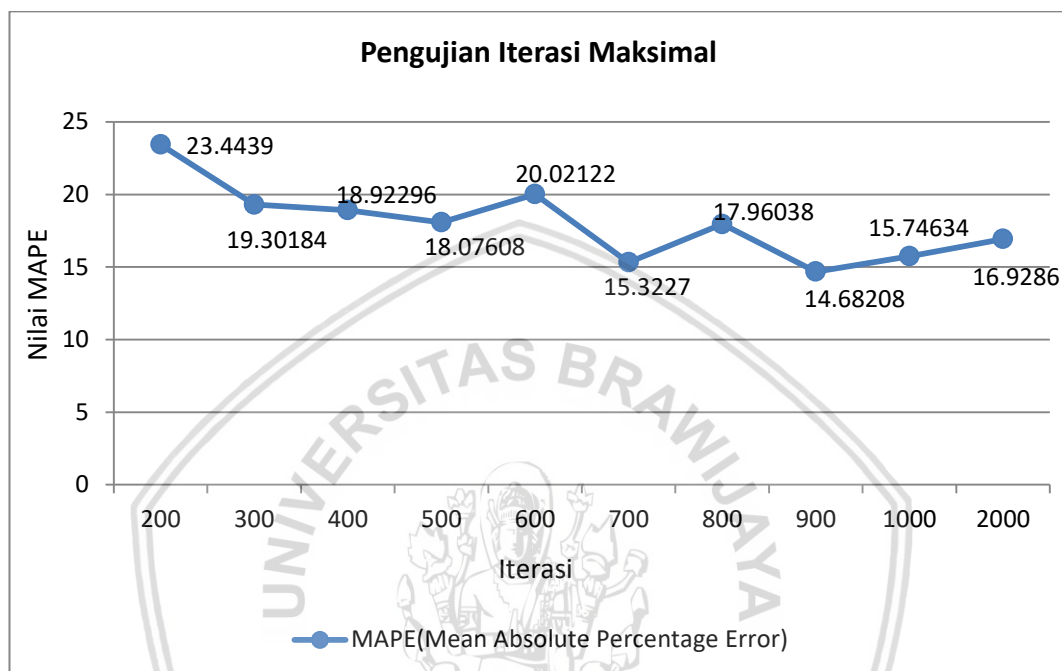
Pada pengujian ini Iterasi maksimal yang di uji diantaranya 200, 300, 400, 500, 600, 700, 800, 900, 1000, dan 2000. Jumlah data latih yang diinputkan sebesar 260 data latih, dengan 59 data uji, nilai learning rate 0.4, dan jumlah *neuron hidden layer* 5. Pengujian dilakukan sebanyak 5 kali percobaan running program, kemudian nilai MAPE dihitung rata-ratanya pada setiap jumlah iterasi maksimal. Pengujian ini bertujuan menemukan batas Iterasi maksimal yang menghasilkan nilai MAPE terendah agar memberikan nilai Akurasi yang baik. Berikut tabel 6.1 hasil dari pengujian Jumlah Iterasi Maksimal.

Tabel 6.1 Hasil Pengujian Jumlah Iterasi Maksimal

Jumlah Iterasi Maksimal	Nilai MAPE (%) pada percobaan ke-n					Rata- rata Nilai MAPE (%)
	1	2	3	4	5	
200	22,5483	24,9346	23,1337	22,5024	24,1005	23,4439
300	23,92	15,8902	22,3959	18,9011	15,402	19,30184
400	13,3498	17,7086	23,8883	15,7855	23,8826	18,92296
500	16,7133	24,8210	17,4606	14,6611	16,7244	18,07608
600	23,8383	21,2128	13,6831	21,4504	19,9215	20,02122
700	14,0132	13,7847	17,0134	14,0975	17,7047	15,3227
800	19,9510	23,2627	15,4788	15,2453	15,8641	17,96038
900	12,9087	13,5194	13,6475	15,8970	17,4378	14,68208
1000	13,4675	17,7953	16,8597	14,0999	16,5093	15,74634

2000	12,8159	13,7385	18,6839	22,7830	16,6217	16,9286
Rata- Rata MAPE (%) Keseluruhan						18,04061

Dari tabel 6.1 jumlah iterasi maksimal yang menghasilkan nilai MAPE terendah adalah 900 dan jumlah iterasi dengan nilai MAPE terbesar adalah 200. Maka jumlah iterasi 900 dapat menghasilkan nilai Akurasi yang baik. Berikut gambar 6.2 representasi grafik pengujian jumlah Iterasi Maksimal.



Gambar 6.1 Pengujian Iterasi Maksimal

dari gambar grafik 6.1 nilai MAPE terendah yaitu 14,68208% dengan jumlah iterasi maksimal sebesar 900. Pada iterasi mulai dari 900 hingga 2000 nilai MAPE tidak terlalu jauh berbeda hingga dapat di simpulkan jumlah iterasi maksimal di mulai dari 900 dapat memberikan nilai MAPE yang baik pada penelitian ini. Pada umumnya jumlah iterasi maksimal yang semakin besar akan menghasilkan nilai output yang mendekati nilai target pada fase pelatihan, namun jika jumlah iterasi terlalu besar waktu running program akan semakin lama, oleh karena itu diperlukan penentuan batas error atau MSE pada inputan agar menemukan jumlah iterasi maksimal yang terbaik dan mempersingkat waktu running program.

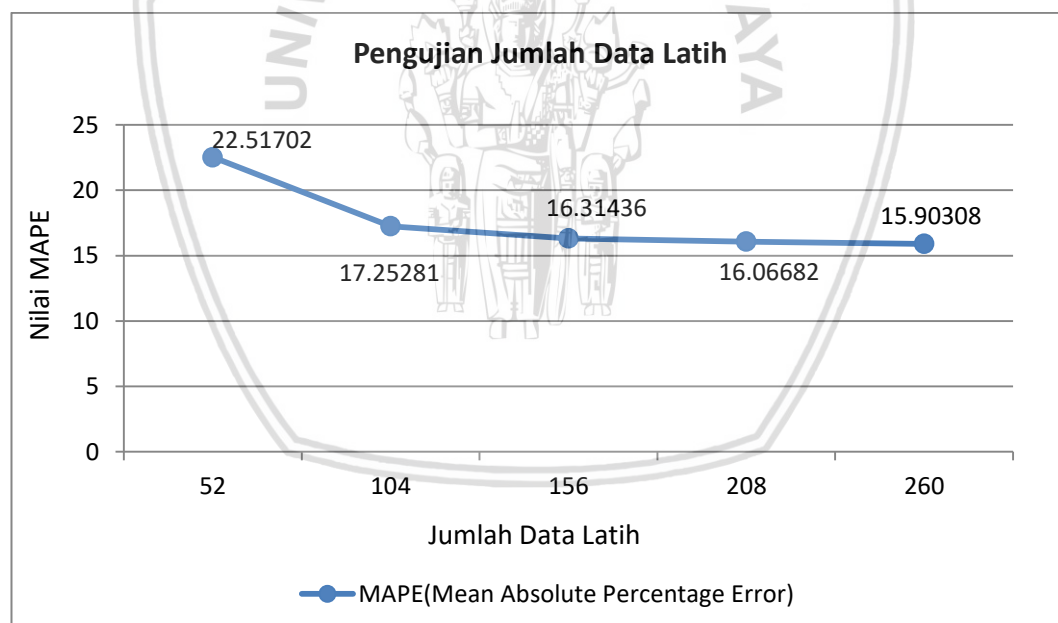
6.1.2 Hasil Pengujian Jumlah Data Latih

Pada pengujian ini jumlah data latih yang di uji diataranya, 52, 104, 156, 208, dan 260. Kelima jumlah data latih ini dikombinasikan dengan nilai learning rate sebesar 0,4 ,jumlah iterasi maksimal 900, jumlah *neuron hidden layer* 5, batas error sebesar 0,001 ,serta 59 data sebagai data uji. Pengujian ini bertujuan mendapatkan jumlah data latih terbaik yang menghasilkan nilai MAPE terendah. Berikut tabel 6.2 hasil dari pengujian Jumlah data Latih.

Tabel 6.2 Hasil Pengujian Jumlah Data Latih

Jumlah data latih	Nilai MAPE (%) pada percobaan ke-n					Rata- rata Nilai MAPE (%)
	1	2	3	4	5	
52	21,1326	23,6098	23,9817	20,5263	23,3347	22,51702
104	22,85526	17,4332	14,7415	17,54809	13,6860	17,25281
156	14,0917	22,4258	15,6660	13,4330	15,9553	16,31436
208	18,8955	13,1482	13,0126	15,3089	19,9689	16,06682
260	12,7091	19,8279	15,7466	15,8191	15,4127	15,90308
Rata- Rata MAPE (%) Keseluruhan						17,61082

Pada tabel di atas dapat dilihat jumlah data latih yang semakin besar menghasilkan nilai MAPE yang semakin rendah. Nilai MAPE terendah dengan jumlah data latih terbesar yaitu 260 dan nilai MAPE terbesar yaitu dengan jumlah data latih terkecil sebesar 52 data latih. Hasil pengujian jumlah data latih direpresentasikan pada gambar 6.2.



Gambar 6.2 Pengujian Jumlah Data Latih

Pada gambar grafik 6.2 nilai MAPE terendah yaitu sebesar 15,90308%. jumlah data latih dengan nilai MAPE yang semakin menurun dimulai dari jumlah data latih sebesar 104 hingga 260. Maka semakin besar data latih pada penelitian ini akan menghasilkan nilai MAPE yang baik.

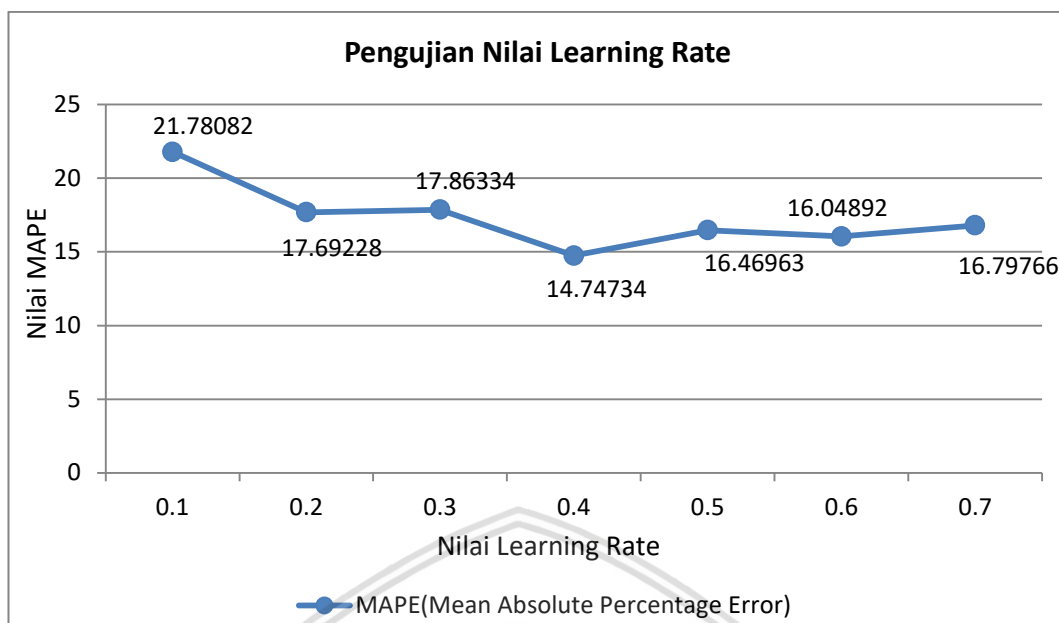
6.1.3 Hasil Pengujian Nilai Learning Rate

Pada pengujian ini nilai learning rate yang di uji diantaranya 0.1 , 0.2, 0.3, 0.4, 0.5, 0.6 dan 0.7. pengujian dipresentasikan dengan nilai MAPE(*Mean Absolute Percentage Error*), dengan jumlah data latih yang diinputkan yaitu 260 data latih, data uji berjumlah 59, maksimal iterasi sebesar 1000, *neuron hidden layer* berjumlah 5 dan batas error sebesar 0,001. pengujian dilakukan sebanyak 5 kali percobaan running program, kemudian dihitung rata-rata nilai MAPE dari 5 percobaan yang dilakukan. Pengujian ini bertujuan mendapatkan nilai learning rate yang menghasilkan nilai MAPE terendah hingga dapat memberikan akurasi terbaik. Berikut hasil Pengujian ditunjukkan pada tabel 6.3.

Tabel 6.3 Hasil Pengujian Nilai Learning Rate

Nilai Learning Rate	Nilai MAPE (%) pada percobaan ke-n					Rata- rata Nilai MAPE (%)	Waktu
	1	2	3	4	5		
0,1	21,3824	23,8983	20,0489	24,701	18,8735	21,78082	24,2 s
0,2	15,8616	19,2965	15,0653	18,2394	19,9986	17,69228	3,4 s
0,3	18,3985	19,9448	15,4931	18,17852	17,3018	17,86334	1,4 s
0,4	13,2498	14,4622	17,5786	12,1929	16,2532	14,74734	2,4 s
0,5	16,5324	14,8033	15,878	15,73646	19,398	16,46963	1,6 s
0,6	16,1891	13,7057	18,8968	16,1188	15,3342	16,04892	2 s
0,7	13,3682	16,9782	19,7979	18,1614	15,6826	16,79766	2,2 s
Rata- Rata MAPE (%) Keseluruhan						17,34286	5,314 s

Berdasarkan Tabel 6.3 Nilai learning rate dengan MAPE(*Mean Absolute Percentage Error*) terendah adalah 0.4, sedangkan nilai learning rate dengan MAPE terbesar adalah 0.1. Maka nilai learning rate 0.4 menghasilkan akurasi terbaik dengan nilai MAPE terendah. Berikut gambar 6.3 Grafik hasil pengujian nilai learning rate.



Gambar 6.3 Pengujian Nilai Learning Rate

Dapat dilihat pada gambar grafik 6.3 nilai MAPE terendah 14,74734% dengan nilai learning rate 0,4 dan waktu running program rata-rata 2,4 detik, sedangkan waktu tercepat yaitu 1,4 detik dengan nilai learning rate 0,3 menghasilkan MAPE 17,86334%. Kisaran nilai learning rate terbaik jika dilihat pada grafik antara 0.4 hingga 0.7 diantara nilai learning rate yang lainnya. Nilai learning rate mempengaruhi laju pembelajaran *backpropagation*, nilai learning rate yang rendah membuat laju pembelajaran lambat, sedangkan nilai learning rate yang terlalu besar membuat laju pembelajaran terlalu cepat hingga tidak memberi hasil yang optimal pada proses pengujian.

6.1.4 Hasil pengujian Jumlah *Neuron Hidden Layer*

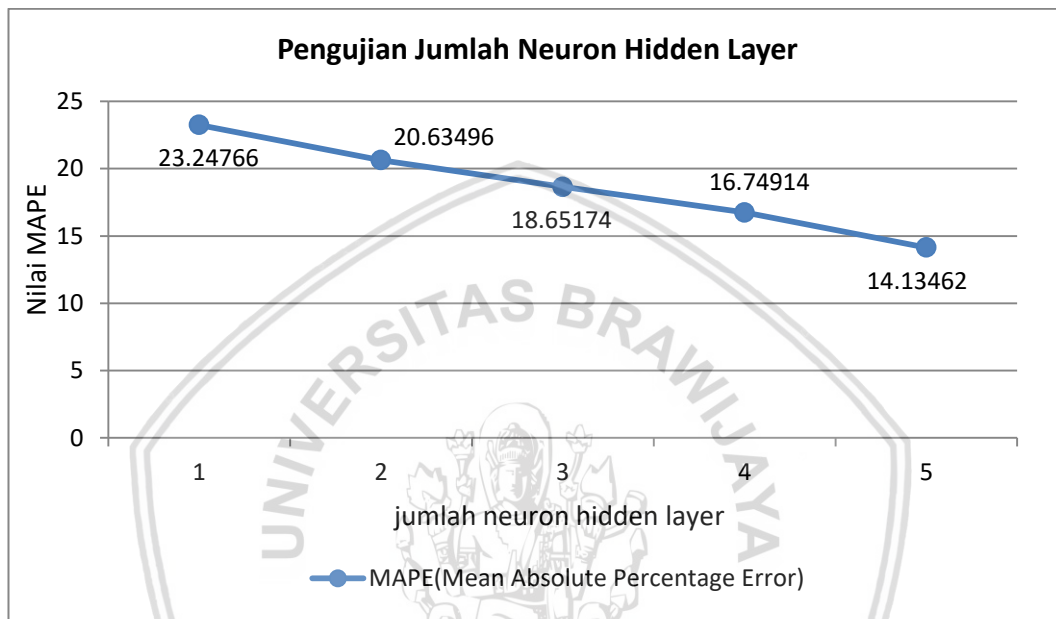
Pada pengujian ini jumlah *neuron hidden layer* yang di uji mulai dari 1 hingga 5. Dengan jumlah data latih sebesar 260 data, nilai learning rate sebesar 0.4, Jumlah iterasi maksimal sebesar 900, batas error 0.001 dan jumlah data yang di uji sebesar 59. Pengujian ini bertujuan menemukan arsitektur *neuron hidden layer* yang terbaik pada proses pelatihan data. Berikut tabel 6.4 hasil pengujian jumlah *neuron hidden layer*.

Tabel 6.4 Hasil Pengujian Jumlah *Neuron Hidden Layer*

Jumlah neuron	Nilai MAPE (%) pada percobaan ke-n					Rata- rata Nilai MAPE (%)
	1	2	3	4	5	
1	23,8551	21,5726	22,8881	24,0018	23,9207	23,24766
2	24,3231	21,1498	22,9902	16,9485	17,7632	20,63496
3	20,1468	16,7063	15,4896	16,5256	24,3904	18,65174

4	18,1492	14,1926	19,6899	15,2939	16,4201	16,74914
5	15,3983	17,4253	13,2993	11,6919	12,8583	14,13462
Rata- Rata MAPE (%) Keseluruhan						18,68362

Pada tabel diatas dapat dilihat nilai MAPE terendah dihasilkan pada jumlah *neuron hidden layer* sebesar 5, sedangkan jumlah *neuron hidden layer* yang menghasilkan nilai MAPE terbesar dengan jumlah 1 *neuron hidden layer*. Berikut gambar grafik 6.4 hasil pengujian jumlah *hidden layer*.



Gambar 6.4 Pengujian Jumlah *Neuron Hidden Layer*

Pada gambar grafik 6.4 nilai MAPE semakin menurun mulai dari jumlah *neuron* sebesar 1 hingga 5, nilai MAPE terendah yaitu 14,13462% dengan jumlah *neuron hidden layer* sebesar 5, maka jumlah *neuron hidden layer* sebesar 5 dapat menghasilkan nilai MAPE yang baik. Hal ini sesuai dengan teori penentuan jumlah *neuron hidden layer* pada bab 2, yaitu jumlah *neuron hidden layer* adalah $\frac{2}{3}$ dari jumlah input layer ditambah output layer.

6.1.5 Hasil Pengujian *K-Fold Cross Validation*

Pada pengujian ini nilai *K-fold* yang di uji adalah 5 dan 10. Pada *K-5* dikelompokkan 66 data setiap fold sedangkan pada *K-7* dikelompokkan 44 data setiap foldnya. Perancangan fold telah dijelaskan pada bab 4, selanjutnya nilai learning rate yang digunakan pada pengujian ini ialah 0.4, iterasi maksimal 900, jumlah *neuron hidden layer* 5 dan batas error 0.001. Tujuan dari pengujian ini ialah menganalisis performa dari model jaringan syaraf tiruan *backpropagation* dan arsitektur jaringan syaraf tiruan yang telah dibuat. Berikut tabel 6.5 dan 6.6 hasil pengujian *K-fold cross validation*.

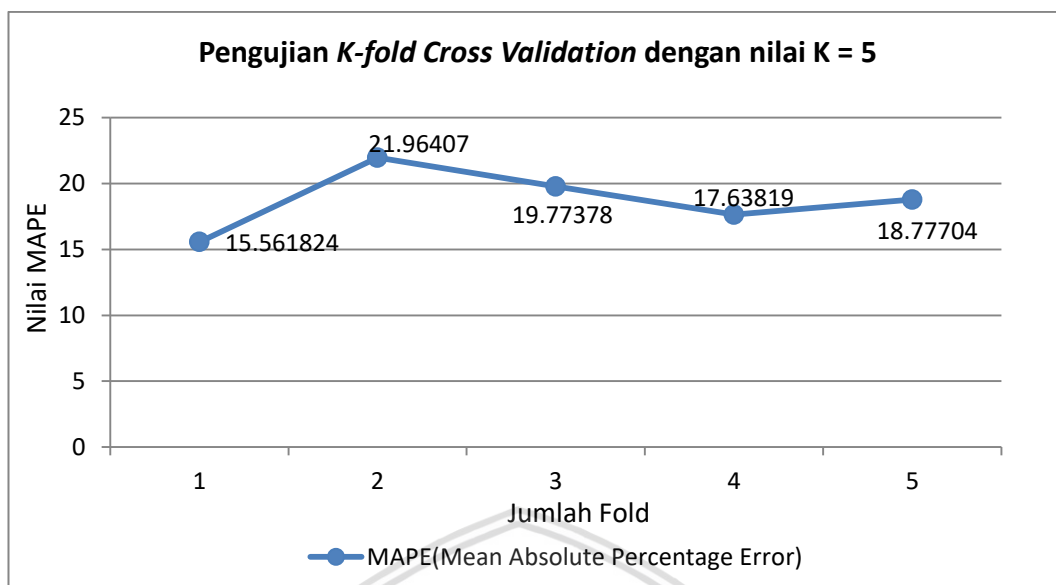
Tabel 6.5 Hasil Pengujian *K-Fold Cross Validation* dengan nilai k=5

Fold	Nilai MAPE (%) pada percobaan ke-n					Rata- rata Nilai MAPE (%)
	1	2	3	4	5	
1	14,20622	12,9966	13,4496	24,7427	12,414	15,561824
2	23,48705	20,2908	21,2671	23,8857	20,8897	21,96407
3	19,4513	19,6177	18,9235	20,9968	19,8796	19,77378
4	19,74405	13,3631	15,7657	16,8658	22,4523	17,63819
5	21,5048	16,9832	12,4118	24,2695	18,7159	18,77704
Rata- Rata MAPE (%) Keseluruhan						18,742981

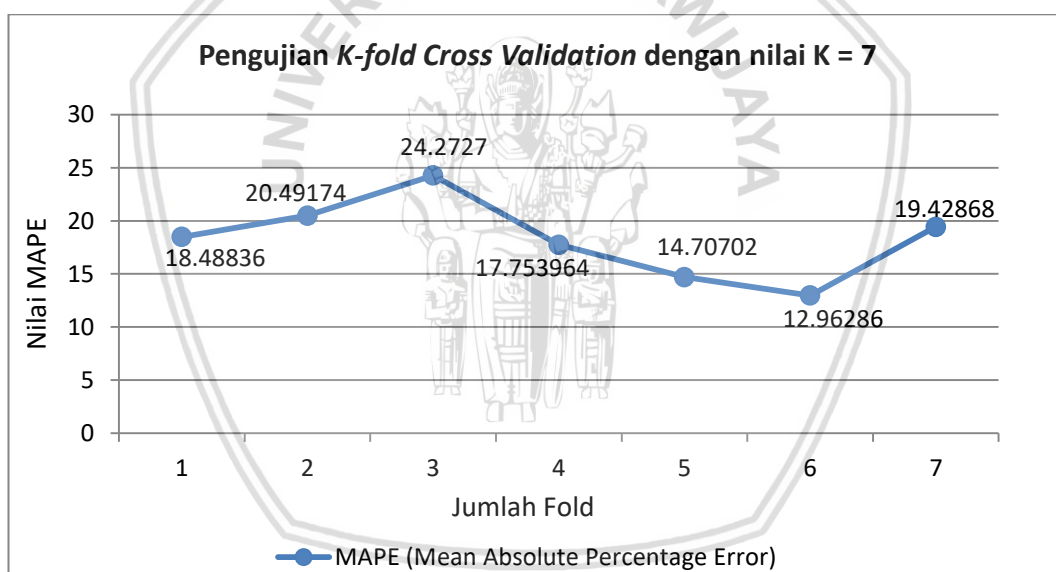
Tabel 6.6 Hasil Pengujian *K-Fold Cross Validation* dengan nilai k=7

Fold	Nilai MAPE (%) pada percobaan ke-n					Rata- rata Nilai MAPE (%)
	1	2	3	4	5	
1	18,2104	18,3293	21,2810	15,631	18,9901	18,48836
2	23,3217	20,8652	15,5629	21,829	20,8799	20,49174
3	24,9253	23,6344	24,8904	24,3353	23,5781	24,2727
4	15,57802	21,6582	22,0562	16,4858	12,9916	17,753964
5	13,1090	12,3232	14,9724	15,0589	18,0716	14,70702
6	13,3936	13,7095	12,3256	12,8498	12,5358	12,96286
7	17,1867	27,5396	17,6597	17,9424	16,8150	19,42868
Rata- Rata MAPE (%) Keseluruhan						18,30076

Berdasarkan tabel 6.5 dan 6.6 pada nilai K-5 nilai MAPE terendah terjadi pada fold ke 1 dan nilai MAPE tertinggi pada fold ke 3. Sedangkan pada nilai K-7 MAPE terendah pada fold ke 6 sedangkan nilai MAPE tertinggi pada fold ke 3. Berikut gambar grafik 6.5 dan 6.6 hasil pengujian *K-fold Cross Validation*.



Gambar 6.5 Pengujian K-fold Cross Validation dengan nilai K= 5



Gambar 6.6 Pengujian K-fold Cross Validation dengan nilai K= 7

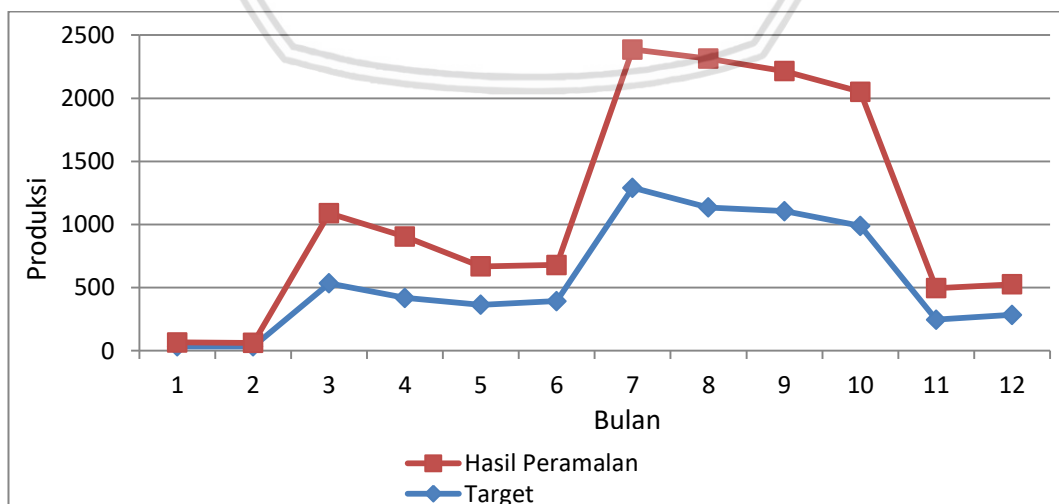
Pada gambar grafik 6.5 dan 6.6 pengujian dengan nilai K-5 menghasilkan nilai MAPE terendah pada fold ke 1 dengan nilai MAPE sebesar 15,561824%, sedangkan pengujian dengan nilai K-7 nilai MAPE terendah pada fold ke 6 dengan nilai MAPE sebesar 12,96286%. Pengujian dengan nilai K-7 menghasilkan nilai MAPE terbaik dikarenakan jumlah data latih yang banyak dan jumlah data uji yang sedikit dibandingkan dengan K-5. Selain itu yang mempengaruhi nilai MAPE pada pengujian ini ialah urutan data yang dijadikan data latih dan data uji.

6.2 Analisis Hasil Pengujian

Setelah melakukan Pengujian Nilai Learning Rate, Pengujian Iterasi Maksimal, Pengujian Jumlah Data Latih, Pengujian Jumlah *Neuron Hidden Layer*, Dan Pengujian *K-Fold Cross Validation*. Didapatkan nilai learning rate terbaik yaitu 0.4, jumlah iterasi maksimal terbaik yaitu 900, jumlah data latih terbaik yaitu 260 dan jumlah *neuron hidden layer* terbaik sebesar 5 neuron. Dari hasil pengujian yang telah didapatkan, parameter tersebut digunakan untuk percobaan peramalan pada data uji sebesar 12 data yang merupakan data produksi kelapa sawit dari bulan januari hingga desember, berikut tabel 6.7 dan gambar 6.7 analisis hasil pengujian menggunakan parameter yang di dapat dari pengujian sebelumnya. Berdasarkan gambar grafik 6.7 dapat disimpulkan hasil peramalan mendekati nilai target dan pola data target dan peramalan menyerupai. Nilai MAPE pada proses pengujian mencapai 10,0047%.

Tabel 6.7 Analisis Hasil Pengujian

No	Target (Ton)	Hasil Peramalan(Ton)
1	33	31
2	33	27
3	533	556
4	417	486
5	362	306
6	392	287
7	1290	1097
8	1134	1180
9	1105	1110
10	988	1063
11	244	251
12	283	243



Gambar 6.7 Hasil Peramalan dengan *Backpropagation*

BAB 7 PENUTUP

Bab penutup menjelaskan hasil akhir penelitian peramalan produksi kelapa sawit menggunakan metode *backpropagation* setelah melakukan pengujian. Kesimpulan akan menjawab rumusan masalah serta memaparkan analisis hasil, kemudian saran akan menjelaskan kekurangan dari penelitian yang dilakukan serta masukan bagi penelitian selanjutnya.

7.1 Kesimpulan

Setelah melakukan penelitian, berikut kesimpulan yang didapatkan untuk menjawab rumusan masalah:

1. Peramalan produksi kelapa sawit menggunakan metode *backpropagation* dengan studi kasus di PT. Sandabi Indah Lestari. Data keseluruhan berjumlah 319 dengan 260 data dijadikan data latih dan 59 data dijadikan data uji. neuron input terdiri atas parameter umur tanaman, jumlah pokok, luas lahan, hasil panen bulan pertama, hasil panen bulan kedua, dan hasil panen bulan ketiga. Jumlah *neuron hidden layer* didapatkan $\frac{2}{3}$ dari jumlah neuron input ditambah neuron output. Neuron output berupa hasil ramalan panen pada bulan keempat dengan skala ton. Metode *backpropagation* dimulai dengan menginisialisasi bobot awal, kemudian fase propagasi maju, fase propagasi mundur dan ubah bobot. Pada tahap inisialisasi bobot awal menggunakan algoritme *nguyen widrow* untuk mendapatkan bobot terbaik yang bertujuan mempercepat waktu pelatihan data.
2. Arsitektur jaringan syaraf tiruan *backpropagation* pada penelitian ini ialah 6-5-1 yang berarti 6 neuron input, 5 *neuron hidden layer* dan 1 neuron output. Berdasarkan hasil pengujian jumlah *neuron hidden layer* yang menghasilkan nilai MAPE terbaik sebesar 14,13% adalah dengan jumlah 5 *neuron hidden layer*.
3. Pada pengujian nilai learning rate MAPE terbaik yang dihasilkan ialah 14,74% dengan nilai learning rate 0,4 serta waktu running program 2,4 detik. Waktu running program yang tercepat yaitu 1,4 detik dengan nilai learning rate 0,3 namun MAPE yang dihasilkan masih cukup tinggi yaitu 17,86%.
4. Berdasarkan hasil pengujian didapatkan nilai terbaik pada parameter iterasi maksimum, jumlah data latih, nilai learning rate, dan jumlah *neuron hidden layer*. Iterasi maksimum yaitu 900, jumlah data latih 260, nilai learning rate 0,4 dan jumlah neuron hidden 5. Kemudian dilakukan analisis hasil dengan menggunakan parameter terbaik yang didapat dari proses pengujian. Pada tahap analisis hasil proses pelatihan menggunakan 260 data latih dan pengujian pada 12 data uji yang merupakan data produksi kelapa sawit dari bulan januari hingga desember tahun 2016, hingga menghasilkan MAPE 10,0047%.

7.2 Saran

Saran menjelaskan kekurangan dari penelitian yang dilakukan serta masukan bagi penelitian selanjutnya, setelah mendapatkan hasil dari penelitian ini penulis memberikan beberapa saran diantaranya:

1. Pada penelitian ini faktor yang mempengaruhi hasil panen kelapa sawit hanya umur tanaman, jumlah pokok, dan luas lahan dan hasil panen pada bulan sebelumnya. Masih banyak faktor-faktor lain yang belum ditambahkan seperti faktor iklim, unsur hara tanah, jumlah tenaga kerja, dan yang lainnya.
2. Diharapkan untuk penelitian selanjutnya dapat memodifikasi algortime *backpropagation* agar mempercepat laju pembelajaran dan meminimalkan jumlah epoch atau menggunakan metode lain yang memberikan akurasi lebih baik dari akurasi yang dicapai pada penelitian ini.



DAFTAR PUSTAKA

- Audela, Edgar. 2015. *Learning Vector Quantization Untuk Prediksi Produksi Kelapa Sawit Pada PT. Perkebunan Nusantara I Pulau Tiga*. S1. Teknik Informatika, USU, Sumatera Utara.
- Azid, I.A. 2003. *Application Of Back Propagation Neural Network In Predicting Palm Oil Mill Emission*. School of Mechanical Engineering Engineering Campus, Universiti Sains Malaysia.
- Badrun, M. 2010. *Tonggak Perubahan Melalui PIR Kelapa Sawit Membangun Negeri*. S2.Direktorat Jendral Perkebunan, Kementrian Pertanian Republik Indonesia, Jakarta.
- Brian, T., 2016. *Analisis Learning Rates Pada Algoritme Backpropagation Untuk Klasifikasi Penyakit Diabetes*. Universitas Trunojoyo Madura Bangkalan.
- Fausett, Laurene V., 1994. *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*. PrenticeHall.
- Hermantoro & Purnawan, W. R. 2009. *Prediksi Produksi Kelapa Sawit Berdasarkan Kualitas Lahan Menggunakan Model Artificial Neural Network*. Agroteknose, 4(2). pp. 1-6.
- Irawan, Eka., 2017. *Analisis Penambahan Nilai Momentum Pada Prediksi Produktivitas Kelapa Sawit Menggunakan Backpropagation*. S2. Magister Teknik Informatika, USU, Sumatera Utara.
- Kemenperin, 2010. *Prospek dan Permasalahan Industri Sawit*. [online] Tersedia di:<[http://kemenperin.go.id/artikel/494/Prospek-Dan-Permasalahan Industri-Sawit](http://kemenperin.go.id/artikel/494/Prospek-Dan-Permasalahan-Industri-Sawit)> [Diakses 1 Agustus 2017]
- Nurhatika, S., 2012. *Sistem Pakar Untuk Mendiagnosis Penyakit Tanaman Kelapa Sawit*. S1. STMIK Atma Luhur Pangkalpina Kepulauan Babel.
- Purnamasari,Ratnaningtyas W., 2013. *Implementasi Jaringan Syaraf Tiruan Backpropagation Sebagai Sistem Deteksi Penyakit Tuberculosis (Tbc)*. S1. Universitas Negeri Semarang.
- Puspita, A, and Unike. 2007. *Penggunaan Jaringan Syaraf Tiruan Metode Backpropagation Untuk Memprediksi Bibir Sumbing*. Seminar Nasional Teknologi.
- Ridhani, M., Putri, R., & Sutopo, S. *Peramalan Dosis Pupuk Berdasarkan Karakteristik dan Lingkungan Tanaman Jeruk Siam Menggunakan Metode Backpropagation*. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, vol. 1, no. 11, p. 1214-1223, juli 2017. ISSN 2548-964X. Tersedia pada: <<http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/412>>. Tanggal Akses: 19 mar. 2018
- Risza, S. 1994. *Upaya Peningkatan Produktivitas Kelapa Sawit*. Yogyakarta. Kanisius.

- Salim Sinaga, interview by Retiana Fadma Pertiwi Sinaga. 2017. Produksi Kelapa Sawit Di PT. Sandabi Indah Lestari (Agustus 27).
- Siang, J. J. 2009. *Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan Matlab*. 98-111.
- Syaban, Kharis & Harjoko, A., 2016. *Klasifikasi Varietas Cabai Berdasarkan Morfologi Daun Menggunakan Backpropagation Neural Network*. S2. FMIPA, UGM, Yogyakarta.
- Wahyuni, Linda & Darma, S., 2014. *Sistem Pakar Mengidentifikasi Gejala Defisiensi Unsur Hara Pada Tanaman Kelapa Sawit*. S1. STMIK Potensi Utama Medan.

